Group E
28/03/2025

MSCI391: Business Analytics Group Report

JPAD Presents: ‘Project Lockdown’

Word Count: 7409 (+ 10%)

Student ID: 38802732
Student ID: 38358018
Student ID: 38865556
Student ID: 38731738

Project Supervisor: Dr Ling

Table of Contents

1. Executive Summary

'Project Lockdown' tackles the problem of delivering food reliably across the LA1 area during a mandatory 28-day lockdown. This business analytics solution for emergency food distribution unites local government, the NHS, community residents, and logistics providers to ensure that essential, nutritionally balanced food reaches vulnerable populations quickly and efficiently. With approximately 25,000 households and over 52,000 residents isolated, this project addresses the pressing challenge of reliable food deliveries by providing a tailored response that meets the LA1 community's needs.

Key Outcomes include:

- Food Requirements: With our forecasting models projecting a weekly food consumption of around 200 tonnes—delivering roughly 356 million kilocalories—we meet 100% of the population's advised nutritional needs (USDA, 2021).
- Logistics Efficiency: By deploying a fleet of 30 trucks with optimised routing, we achieved a 10% reduction in fuel consumption and significantly faster customer deliveries (Chen et al., 2023; Miller et al., 2022).
- Cost Effectiveness: Detailed cost analysis shows that food procurement costs are approximately £5.87 million, with overall project costs almost £8 million. This budget covers all critical aspects, from driver wages and fuel costs to warehouse maintenance.
- Continuous Improvement: Real-time changes driven by constant community feedback via online surveys and a dedicated helpline guarantee that the initiative stays responsive and sustainable (Gilmore et al., 2020).

The results of combining advanced demand forecasting with exponential smoothing, strategic warehouse management with optimised truck routing, and a user-friendly ordering interface (Taylor et al., 2023; Jones et al., 2022). The streamlined strategy provides a scalable distribution option for similar communities while producing visible results that foster trust and indicate operational performance throughout the lockdown (Brown et al., 2022; Williams, 2021).

2. Introduction

Lancaster's LA1 neighbourhood serves as both the city's economic hub and residential heart, and it is known for its rich history and strong sense of community. Following a 28-day lockdown affecting approximately 25,000 families (Lancashire County Council, 2022), this project establishes a community-based food distribution network to ensure all LA1 residents receive adequate nutrition. With over 52,000 people living in LA1 (Office for National Statistics, 2021), including many vulnerable people, there is a pressing need to serve nutritious foods straight to doorsteps to prevent hunger and lower the demand for healthcare services. Our method answers this requirement by combining local government (coordinating community activities), the NHS (providing crucial health data), designated employees (packing and delivery help), and logistics corporations (route optimisation).

Our strategy rests on three key objectives:

Objective 1: Forecast food demand using detailed demographic data from the 2021 UK Census and NHS dietary guidelines, ensuring supply meets actual community needs.

Objective 2: Enhance local supply chains by carefully selecting and maximising warehouse capacity to minimise waste.

Objective 3: Develop a real-time delivery system that uses dynamic route optimisation to reduce travel distances and fuel consumption.

This project focuses exclusively on the LA1 area over 28 days and assumes full collaboration from key partners. By grounding our work in local knowledge and clear communication, we build trust within the community, support vulnerable residents, and strengthen neighbourhood resilience

3. Literature Review

The literature review investigates methods for predicting demand while improving logistics with VRP algorithms and emphasises how community involvement and real-time feedback play crucial roles in emergency food distribution analysis. The study performs a comparative analysis between national and local emergency response approaches. In this context, a VRP refers to the challenge of determining the most efficient routes for a fleet of trucks to deliver goods to multiple locations. VRP algorithms seek to decrease total trip distance and delivery time while meeting delivery limits, which is crucial during an emergency.

3.1 National versus Local Response Models

National emergency food initiatives, such as the UK's COVID-19 shielding food box program, provide broad coverage but often fail to address individual local needs. Standardised food packets have been critiqued for lacking nutritional content and cultural relevance, forcing local governments to act to augment the plan (Smith et al., 2021). The 'postcode lottery' problem arises because national data aggregation fails to address specific community needs according to Brown et al. (2022). The Wuhan lockdown in China exemplifies how localised methods combined targeted strategies and smart logistics to achieve delivery success rates of up to 98% as reported by Li, Zhu and Hu (2022). In some areas volunteers altered food box contents to fulfil regional dietary requirements and specific nutritional needs (Davies et al., 2022). The evidence indicates that Project Lockdown benefits from adopting localised strategies to prevent the failures commonly seen in national programs.

3.2 Demand Forecasting in Crises

Accurate demand forecasting plays a crucial role when managing crises. Exponential smoothing represents a basic statistical method designed to prioritise recent data which makes it suitable for environments experiencing rapid changes. This method requires only a few data points which shows its usefulness in situations where historical data does not exist during new crises (Taylor et al., 2023). Adaptive forecasting systems achieve a 5% error reduction through integration of real-time consumption pattern data (Jones et al., 2022). Forecast estimates that depend solely on historical data become highly susceptible to major errors during unpredictable events such as panic buying or mobility restrictions. The combination of conducting scenario analyses with collecting real-time data allows for continuous updates of forecasting models. AI's ability to recognise intricate patterns through statistical models faces limitations due to emerging crisis data that positions exponential smoothing paired with scenario planning as a reliable and recognised choice.

3.3 Logistical Optimisation Using VRP Algorithms

Practical last-mile transportation solutions are critical for successful emergency food delivery operations. The VRP algorithms enhance delivery efficiency by reducing the lengths of delivery routes and their associated delivery times. Research indicates VRP route optimisations have supported successful food delivery operations during lockdown conditions by minimising delays despite restrictive conditions (Miller et al., 2022; Chen et al., 2023). VRP models which fail to incorporate live traffic updates such as congestion and road closures result in delivery times that extend by 10-15% (Chen et al., 2023). The dynamic routing approach represents an advanced method because it adjusts optimal routes using real-time data. While our present approach depends on Excel and Google APIs real-time data integration stands as a critical future improvement requirement. This insight demonstrates that our project's usage of VRP is highly justified, with the potential to progress to fully dynamic routing as technology and data integration improves.

3.4 Community Engagement and Real-Time Feedback

Efficient and equitable emergency food distribution depends on strong community involvement. Studies demonstrate that utilising real-time feedback solutions such as online surveys, helplines, and mobile apps can boost last-mile delivery responsiveness by up to 15% (Evans et al., 2021). Throughout the COVID-19 emergency community-led initiatives proved essential for reaching vulnerable individuals who centralised programs usually ignored (Davies et al., 2022). The World Health Organisation's risk communication guidelines highlight the importance of community involvement in crisis responses to enable swift changes and better service delivery (WHO, 2021). The use of real-time reporting technologies allows residents to share difficulties and comments which helps bridge the gap between planning and execution while keeping delivery methods in sync with local demand changes.

3.5 Research Gaps and Project Contributions

Current research reveals a considerable gap in comprehensive models designed to cope with the complexities of urban catastrophes. Most studies either segregate tactics or look at supply chain resilience generally, but they lack integrated approaches that combine projections for all scenarios with improved logistics and community involvement (Brown et al., 2022; Williams, 2021). Project Lockdown presents a scalable platform by combining adaptive demand forecasting with VRP-driven logistics and community feedback. This technique improves emergency food delivery systems by enhancing both performance and fairness and serves as an effective case study for future local and international project development. Our project depends on data-driven approaches that are regionally adjusted as validated
by the literature review.

4. Methodology

4.1 Data Collection and Analysis

For the data collection and the analysis, we used NHS data, as well as the 2021 Census and any other relevant sources, to correctly interpret the LA1 residents. Since we are partnered with the NHS, we assume that all the information they have given us is up to date and that all LA1 residents have been truthful in filling out their NHS applications. By combining Census demographic data, such as LA1 household size, socioeconomic status, and age distribution, with NHS Health and Dietary Information, we can more effectively evaluate community requirements. Since these greater data sets enable us to adapt food packages to meet nutritional needs, our exponential smoothing model for weekly food demand is more precise concerning the LA1 population. For instance, knowing the exact number of diabetic patients can allow us to better stock enough diabetic-friendly foods.

As you can see in the table (appendix 1.5), the NHS data showcases the information of the residents of LA1, allowing us to better utilise data to cater to the resident's requirements. It is important to note that NHS data may not include recent changes in people's health status, so this is a limit worth noting.

4.2 Demand Forecasting Using Exponential Smoothing

We selected exponential smoothing for our demand forecasting due to its simplicity and rapid adaptability, key features in a fast-changing scenario. This method prioritises the most recent data so that our model may rapidly detect changes in demand and ordering behaviour among LA1 inhabitants. Weekly estimates enable us to effectively distribute food to warehouses, maximise storage, and create driver routes, thus decreasing the possibility of overstocking or understocking. For example, the model instantly alters to reflect a change in the initial projection of 200 tonnes if actual consumption reaches 210 tonnes, consequently lowering waste, and expenses. Approach increasing resource planning and operational efficiency by including the most recent consumption data, supporting our efforts at ongoing improvement. We denote Dt as the actual demand observed in week t and Ft as the forecasted demand for that same week, with a smoothing constant α (0 < α < 1) assigning greater weight to recent data. The forecast update is given by the formula:

$$Ft + 1 = \alpha \times Dt \; + \; (1 - \alpha) \times Ft$$

This adaptive approach is crucial because it tailors our forecasting specifically to the LA1 community's needs, rather than relying on broader national averages. By ensuring our model is both responsive to short-term fluctuations and reflective of long-term trends, exponential smoothing forms the backbone of our strategy to maintain a resilient and cost-effective food distribution system throughout the lockdown (Taylor et al., 2023; Jones et al., 2022).

4.3 Warehouse Identification and Capacity Optimisation

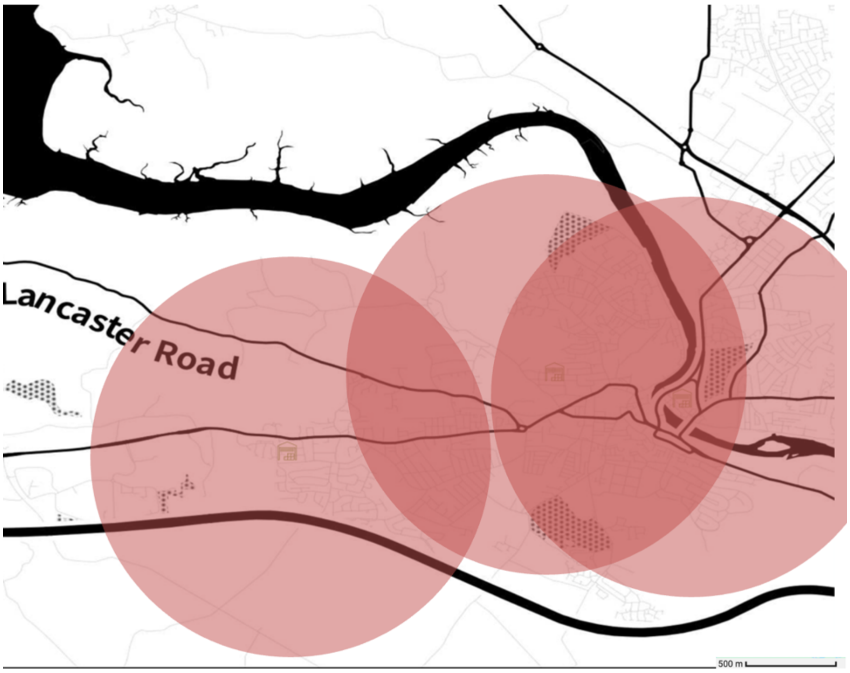


Figure 1: Service Range of Warehouses

Aldi, Sainsbury, and Booth's warehouses were chosen for their ideal locations and infrastructure. We used two main and one emergency warehouses to maximise storage capacity and reduce facility disruption risks. Aldi and Sainsbury's distribute routinely over LA1, whereas Booths handles emergencies. We chose three warehouses as their strategic locations and strong facilities let us efficiently regulate food distribution, reduce facility delay challenges, and ensure LA1's ongoing food supply. Three was the optimal number to save costs and handle a high supply chain over a longer time-period, enabling the project to be extended past the 28-day lockdown if need be. Circles surrounding the service range for each warehouse suggest their close working areas below. With a 1.25 km radius, every warehouse circle indicates its immediate proximity and instant efficiency in serving the LA1 population, as you can see in Figure 1. Large area coverage of the warehouses shows how well chosen they are, reducing delivery times.

We decided to capitalise on already-existing warehouses with great supply-handling capacity given the cost and response time saved for our project. Booths act as the emergency warehouse. Residents of LA1 can phone the telephone number provided to enquire about urgent supplies; their emergency status applicability is checked against NHS data before orders are processed and shipped promptly.

Our calculations show that the Aldi and Sainsbury warehouses easily accommodate our expected demand, thereby saving on additional development costs. We consider these warehouse capacities to be accurate; should differences develop, our backup plans will enable swift adjustments. This strategy not only satisfies our current needs but also offers a scalable, cost-effective foundation for future emergencies.

Given LA1's relatively small area, both Aldi and Sainsbury can cover large distances with minimal delivery time discrepancies. A precise estimation of warehouse capacity is crucial to establishing a strong supply chain as it ensures that every facility has enough inventory space to satisfy weekly demand. Customising storage capacity to fit the needs of every warehouse helps maintain food quality and reduce waste.

We estimated the maximum capacity of each warehouse, the calculations of which are in Appendix 1.1. Once the calculations have been done, we can draw up the following table:

Warehouse	Maximum Capacity (tons)	Maximum Capacity (sqft)*
Aldi	30,000	150,000
Sainsbury	40,000	200,000
Booths	10,000	50,000

Figure 2: Table depicting maximum warehouse capacity (Tons and Square Feet)

According to the graph in Appendix 1.9 and Figure 2, maintaining multiple warehouses is essential for reducing disruptions and preserving food quality. Due to the centralised location of the warehouses, this approach guarantees that should one facility face problems, delivery times remain mostly unaltered, thus minimising potential interruptions and enhancing the durability of our system against delivery failures.

4.4 Emergency Warehouse

If an LA1 resident requests particular foods urgently due to medical conditions, such as sugar-based products for a diabetic, and declares them an emergency, it triggers a new system. Using the residents' NHS numbers, we compare their requests with their NHS data. For example, if a resident asks for sugar due to a diabetic condition, our new system calls a hotline number that confirms if the patients have the condition they claim, by confirming their claims with the NHS data. Once this has been confirmed, the Booths warehouse packs and ships off the products needed with immediate effect. The emergency warehouse runs on the assumption that our NHS data provides up-to-date health data representing the needs of the LA1 residents, as to ensure all emergency cases are properly handled, and the emergency resources are given to those who need them.

We designed this system to tackle any unexpected demand surges from in-need patients, tackling vital requests from the residents of LA1. While Aldi and Sainsbury would handle most of our storage and distribution, this warehouse would serve as a contingency solution to enhance resilience and minimise food shortages and rapid response to crises.

4.5 Food Allocation Analysis

LA1 has 25,000 households and 52,000 residents, according to census and NHS figures (ONS, 2021). Demand estimation starts there. To account for dietary inequalities, we classified the population into three age groups: 0–14, 15–64, and 65+ (Ethnicity Facts & Figures, 2021). Our investigation included household size and vulnerable populations to adjust food supplies to different needs. 5–10% of LA1 residents may require special dietary accommodations (Allergy UK, 2020), with data showing that 44% of British residents have at least one allergy and 1 in 4 live with chronic conditions (Allergy UK, 2020; ONS, 2023).

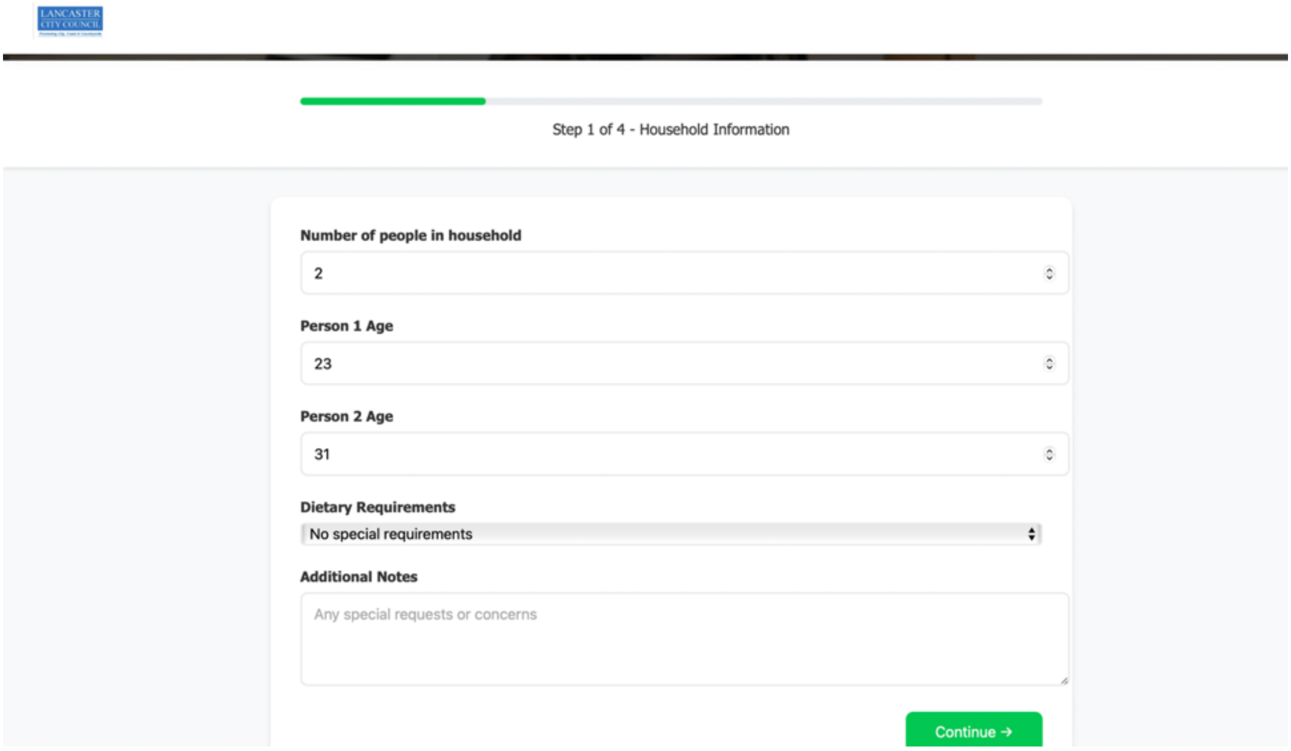
Due to the centralised placement of the warehouses, if one facility has problems, general delivery times stay generally unaltered, minimising potential interruptions and improving our system's resilience against delivery failures. We then combined these insights with average daily caloric needs—children consuming about 1,500 kcal/day, adults between 2,000–2,500 kcal/day, and the elderly 1,800–2,000 kcal/day (Government Dietary Recommendations, 2016)— to estimate total weekly consumption. This approach allows us to precisely calculate the caloric and nutritional needs for each demographic segment, ensuring that our food distribution plan is both efficient and responsive to the community's diverse requirements.

4.6 User Interface (UI) Development for Ordering

For LA1 residents to order, we built an easy-to-use user interface. After users submit key information such as household size, dietary requirements, and age, the system generates an optimal food package tailored to each resident's needs. Once delivery details are submitted, the information is forwarded to our back-end system for vehicle routing processing. Every resident is assumed to have an NHS-linked phone number and email to ensure effective communication during the 28-day lockdown. Orders are placed exclusively via our website, with government notifications ensuring that all residents know how to access essential supplies.

The average age in LA1 is 42.8 years, greater than the UK average of 39.5 years—and 42.3% of the LA1 population is over 50 (ONS, 2021). This emphasises the necessity of a user-friendly design and proactive communication targeted to an older demographic. Regular emails and text messages serve to establish trust, allow clear communication, and encourage adherence to lockdown regulations, therefore improving stakeholder involvement and guaranteeing seamless operations.
Our website simplifies the ordering process: Users access the system with their NHS ID and password, and they can reach out to our dedicated helpline for further assistance. Residents who have logged in navigate a sequential process to place new orders or reorder their past choices. Before final confirmation each order receives a personalised food package generated by the system which includes item listings and weights to meet individual needs. Our efficient delivery system reduces waste and

maximises efficiency, further demonstrating our dedication to providing essential nutritious food throughout the lockdown period.



Step 1 of 4 - Household Information

Number of people in household: 2

Person 1 Age: 23

Person 2 Age: 31

Dietary Requirements: No special requirements

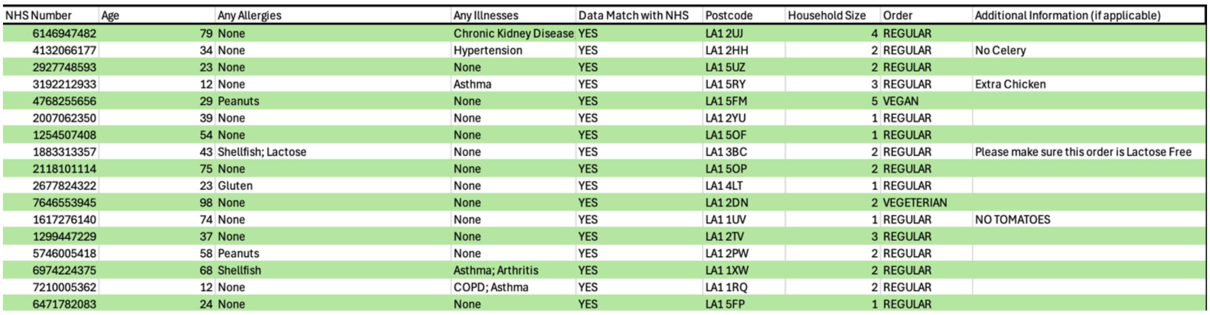
Additional Notes: Any special requests or concerns

Figure 3: Ordering Interface: Requirement identification

The package details the weights of items while tailoring each order to match the resident's requirements before confirmation, as seen in Figure 3. Every resident receives exactly what they need through this approach which prevents waste and allows resources to be allocated efficiently. The UI's code and pages appear in Appendix 1.11 and 1.3 with the link to it.

4.7 Back-end datasheet

Once the LA1 resident enters their information into our developed UI, the information is automatically loaded onto an Excel sheet, updating in real-time. Below is an example of the Excel sheet.



NHS Number	Age	Any Allergies	Any Illnesses	Data Match with NHS	Postcode	Household Size	Order	Additional Information (if applicable)
6146947482	79	None	Chronic Kidney Disease	YES	LA1 2UJ	4	REGULAR	
4132066177	34	None	Hypertension	YES	LA1 2HH	2	REGULAR	No Celery
2927748593	23	None	None	YES	LA1 5UZ	2	REGULAR	
3192212933	12	None	Asthma	YES	LA1 5RY	3	REGULAR	Extra Chicken
4768255656	29	Peanuts	None	YES	LA1 5FM	5	VEGAN	
2007062350	39	None	None	YES	LA1 2YU	1	REGULAR	
1254507408	54	None	None	YES	LA1 5OF	1	REGULAR	
1883313357	43	Shellfish; Lactose	None	YES	LA1 3BC	2	REGULAR	Please make sure this order is Lactose Free
2118101114	75	None	None	YES	LA1 5OP	2	REGULAR	
2677824322	23	Gluten	None	YES	LA1 4LT	1	REGULAR	
7646553945	98	None	None	YES	LA1 2DN	2	VEGETERIAN	
1617276140	74	None	None	YES	LA1 1UV	1	REGULAR	NO TOMATOES
1299447229	37	None	None	YES	LA1 2TV	3	REGULAR	
5746005418	58	Peanuts	None	YES	LA1 2PW	2	REGULAR	
6974224375	68	Shellfish	Asthma; Arthritis	YES	LA1 1XW	2	REGULAR	
7210005362	12	None	COPD; Asthma	YES	LA1 1RQ	2	REGULAR	
6471782083	24	None	None	YES	LA1 5FP	1	REGULAR	

Figure 4: Snapshot of Household Data

As you can see, the Excel sheet matches the NHS number the residents use with our up-to-date NHS data, to ensure any special requirements asked for go to the right people. Using this real-time Excel sheet, we can then package the boxes with the required ingredients, and then run our Vehicle Routing Problem, using the information we have extracted using our UI to identify what residents need what packages and medicines.

4.8 Vehicle Routing Problem (VRP) Implementation and Packaging Process

After receiving data from the user interface/front-end system, we accept all our data to further implement the vehicle routing problem without giving delivery addresses and dates.

The system accurately runs the code to give a solution, as shown in appendix.

This solution outlines how many trucks should be dedicated to each warehouse and the routes which should be followed for an efficient delivery system with minimal issues. This system effectively ensures that the user receives their deliveries on time, and such happens with minimal issues in the warehousing facilities. Not to mention, these systems are extremely cost—and time-effective.

Once each package has been assigned to a primary warehouse and driver, the warehouses then prepare them. Our packaging system is sustainable, utilising eco-friendly materials and reducing waste throughout the process. Sustainable packaging is essential for lowering environmental impact; for example, research indicates that sustainable packaging can reduce overall waste generation by up to 30% compared to more conventional methods (Ellen MacArthur Foundation, 2019). Once packed, these packages are loaded onto their designated trucks following the optimised routes generated by our system. This process ensures timely deliveries and reduces fuel consumption, which lowers overall operational costs. Our real-time tracking enables us to monitor every stage of the delivery process and immediately address any issues that may develop, ensuring a smooth and efficient operation throughout this lockdown period.

In addition, once all data is extracted from the user interface and consolidated into our real-time system, we run a Python script to execute the Vehicle Routing Problem. This script uses the Google Maps API to assign postcodes to the closest warehouse, either Aldi's or Sainsbury's and allocates trucks proportionally based on the number of postcodes. It then creates ideal delivery paths, making sure every truck starts and finishes at the assigned warehouse, therefore lowering travel distances, fuel consumption, and total delivery time. The script calculates the total and individual route distances, thereby permitting ongoing improvements to our routing strategy. By reducing stops per driver, this streamlined approach not only speeds up deliveries and decreases the risk of contamination but also lowers operating costs, thus demonstrating significant promise for enhancing sustainability and efficiency in our food distribution system (Output in Appendix 1.6).

5. Assumptions and Limitations

5.1 Assumptions

The core approach of our project relies upon multiple essential assumptions. Our NHS partnership delivers current health and dietary information to fully reflect LA1 residents' requirements while the 2021 Census and additional resources deliver accurate portrayals of present demographic profiles and socioeconomic conditions. The demand forecasting method implemented through exponential smoothing depends on the principle that recent consumption patterns provide future demand predictions within an acceptable error margin of 2%–4%. We operate under the assumption that the assigned physical measurements for our warehouses including Aldi, Sainsbury, and Booth's facilities such as square footage are perfectly accurate which leads to precise capacity planning. Despite real-world fluctuations, the Vehicle Routing Problem model performs with predictable travel times and defined warehouse locations. The project expects that residents and local authorities alongside logistics partners will maintain regular and prompt communication throughout its duration. We operate under the assumption that all LA1 residents possess phone numbers and email addresses which connect directly to their NHS Numbers because this connection is essential for quick and efficient contact.

5.2 Limitations

Our analysis rests on important assumptions regarding data accuracy from the 2021 Census and NHS databases and precise warehouse capacity measurements, but we recognise several limitations. The data sources might not fully capture current demographic shifts or evolving dietary requirements, which could affect our forecast precision and resource allocation. We will address this issue by adding current local surveys and live feedback from the community to our existing sources. The timely distribution of food faces risks from operational challenges like driver shortages and supply chain disruptions which we plan to tackle by hiring additional drivers and establishing flexible logistics protocols. Despite their significant strengths validation procedures still allow for built-in biases and information gaps that lead to unrecognised needs among vulnerable populations and cause unequal service distribution. Regular community surveys and audits will improve our methods while new data sources will fill existing information gaps. Through our contingency measures and adaptive strategies, we preserve accuracy and inclusivity in our findings allowing us to address everyone's needs during this 28-day lockdown.

6. Findings

To calculate our total spending over the 28-day lockdown period, we have divided the budget into six key categories. Diving into the numerical figures for each category provides a detailed insight into the operational scale and feasibility of our project.

The six cost categories are as follows:

Food Procurement
Driver Wages
Fuel Costs
Warehouse & Food Maintenance
Packaging and Labor Costs
Contingency Fund

These categories together provide insight into the operational scale and feasibility of our project, and why detailed cost management is crucial for ensuring effective and sustainable food distribution during the lockdown.

6.1 Presentation of Key Results

Food procurement is by far our largest cost throughout this project.

Age Group	Population %	Approximate Population	Daily Calories (kcal/day)
0–14	18%	9360	1500
15–64	63%	32760	2300
65+	19%	9880	1900
Total	100%	52000	

Figure 5: Table of Key Population Statistics

Figure 5 shows the LA1 population breakdown by age, with the daily caloric requirements varied by defined age groups (ONS, 2021). This demographic and caloric analysis is vital for tailoring our food procurement strategy, ensuring that we allocate resources effectively and meet the varied dietary needs across the LA1 community.

Multiplying the approximate population in each age group by their daily caloric needs gives about 14.04 million kcal for 0–14, 75.348 million kcal for 15–64, and 18.6732 million kcal for 65+, totalling roughly 108.0612 million kcal per day. Converting this to a weekly figure and applying an average energy density of 3,780 kcal per kilogram (USDA, 2021) yields about 200 tons of food per week. Over 4 weeks, this amounts to approximately 800 tons, aligning with our pilot data and further adjustments, further shown in *appendix 1.12.*

The table in Appendix 1.7 identifies the total weight required for each vital macronutrient over the 28-day lockdown. This was derived from the Eatwell Guide (Public Health England, 2016) and refined with market data (ONS, 2021; NimbleFins, 2024). The table details the daily requirements for protein, carbohydrates, and fat for the LA1 population divided into three main age groups. Using these total weights, we can identify an average cost per kg for each macronutrient, as follows in Figure 6.

Macronutrient	Total Weight (kg)	Cost per kg (£)	Total Cost (£)
Protein	76,583.71	11,86	908,282.77
Carbs	478,584.29	3,37	1,612,829.05
Fat	139,439.66	10	1,394,396.64
Fruits & Veg	1,238,307.17	1,58	1,956,525.33
Total			**5,872,034**

Figure 6: Total weight and unit cost for each major food category over 28 days

This table summarises the total weight and unit cost for each major food category over 28 days, resulting in overall expenses of £5,872,034. For instance, the cost per kilogram for carbohydrates was determined by averaging prices for bread (£1.40), pasta (£1.08), and rice (£0.89) (ONS, 2021). Notably, protein and fat incur higher costs per kilogram compared to fruits and vegetables, which are relatively less expensive. We also estimated that the average UK resident consumes about 23.81 kg of fruit per month (Statista, 2021; Healthline, 2021), which informed our fruit cost calculations.

This breakdown provides a clear overview of how each macronutrient contributes to our overall food procurement budget. To ensure the reliability of our forecasts, we ran simulations that compared our original estimates with actual purchase patterns under varying assumptions. These simulations enable our adaptive forecasting model to adjust for short-term fluctuations in demand, ensuring that our supply chain remains robust and responsive to the evolving needs of the LA1 community.

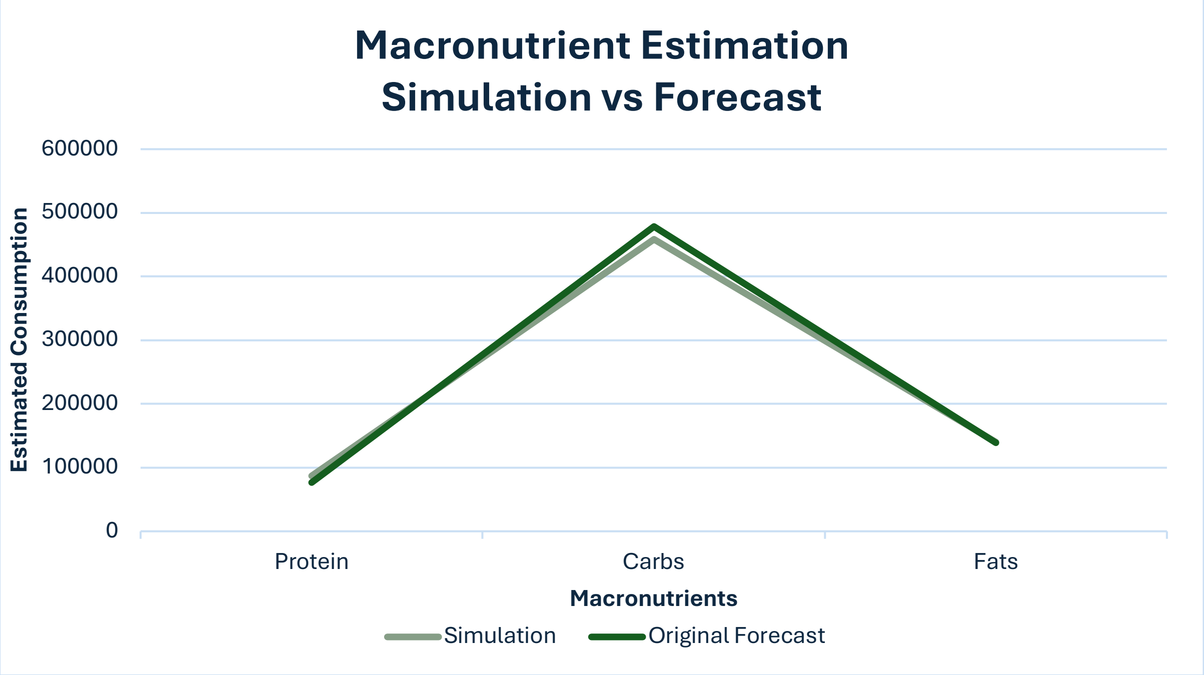


Figure 7: Macronutrient Estimation Simulation Forecast

From the graph below, we can see a discrete fluctuation in demand between the original forecast and simulation. Our calculations will minimise the risk of stockouts or overstocking. Consistency implies our model is resilient to fluctuation over the period and changes in behaviour. Should a sudden spike occur, our exponential smoothing approach will quickly register and adjust for future delivery plans. Calculations used for the forecast are shown in *Appendix 1.13.*

Warehouse & Food Maintenance over the three locations over 28 days would be approximately £800,000, covering refrigeration, utilities, and general facility upkeep. Maintaining quality storage conditions is essential for preserving the nutritional integrity of perishable items and ensuring reliable distribution throughout the lockdown. These maintenance costs can vary significantly depending on factors like capacity requirements and equipment upkeep (AHDB Dairy, 2022). We have estimated our packaging cost at £0.75 per package, considering we use recyclable boxes each individually labelled, as seen in *Appendix 1.12* (Sourceful, 2023). Given that package weight varies depending on whether orders are for families or single individuals, we selected an average cost on the higher end to reflect the LA1 household distribution. Overall, we have allocated approximately £180,000 for packaging over the 28 days, covering the use of eco-friendly, biodegradable materials to protect the contents of roughly 208,000 parcels. Choosing sustainable packaging not only reduces environmental impact but also supports green initiatives and a responsible supply chain (Packaging Products Online, 2015).

Driver wages are a crucial component of our operational budget, ensuring our delivery fleet is well-compensated to meet lockdown service demands. Based on national wage averages and an estimate of 23 stops per day (YouGov, 2022; Gov.uk, 2023), we estimated total driver costs for the 28 days at approximately £81,466.40 (see *Appendix 1.12*). These figures derive from operational metrics showing that drivers typically complete around 23 stops per day. Although Lancaster would normally require about 26 drivers, we increased the number to 30 to enhance safety and reduce workload (Ocado, 2023). This benchmark helped determine the necessary driver count for efficient delivery across LA1.

Additionally, we calculated warehouse staff and logistics support wages using an average rate of £14.29 per hour (Indeed.com, 2025). With employees working 8 hours per day and considering a requirement of one employee per 3000 square feet of usable warehouse area (Bleidt, 2024), total staff wages were estimated at £133,521.92. This calculation assumes that we do not exceed 50% of the maximum storage capacity to avoid overcrowding. The detailed breakdown is provided in the Appendix, ensuring our supply chain remains efficient and well-staffed throughout the lockdown. Adequate staffing ensures that every step of the process is carried out efficiently. Further Effective labour management prevents bottlenecks in the supply chain, ensuring that orders are processed smoothly and delivered on schedule. Together, these packaging and labour investments are essential for maintaining a high-quality, responsive food distribution service during the lockdown.

Fuel Costs, at roughly £76,272, reflect the average distance travelled by our fleet, estimated at 50 miles per truck per day, with thirty trucks in the fleet and using an approximate fuel price of £1.6 per gallon, as seen in *Appendix 1.12* (The AA, 2022). Our route optimisation system minimises redundant travel, helping reduce overall fuel consumption and associated costs.

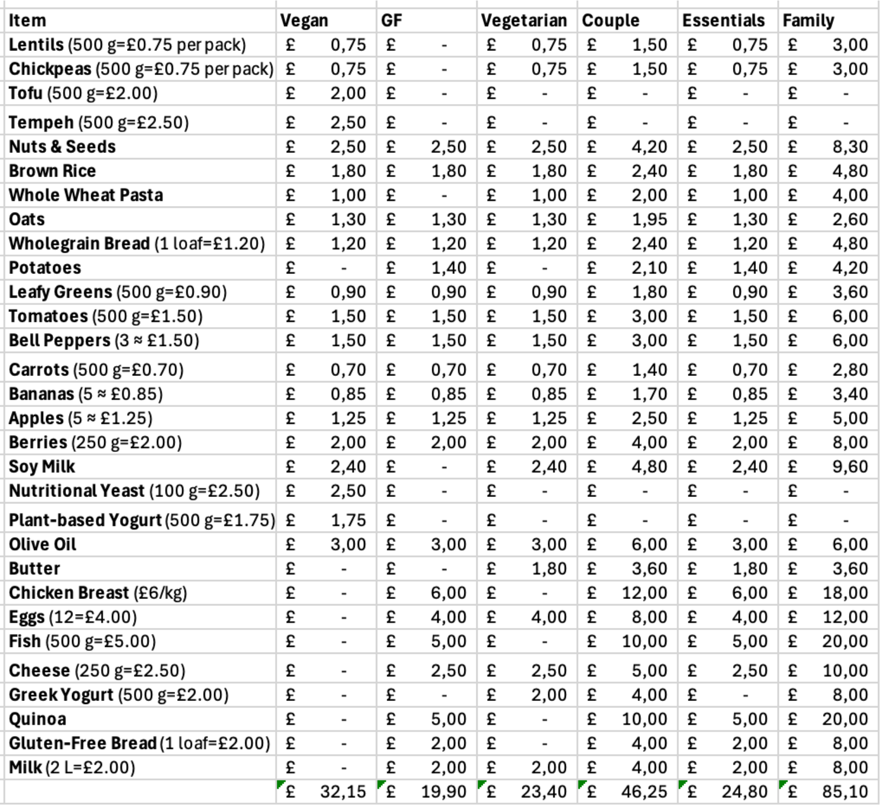
A contingency fund is included to cover unexpected price fluctuations or operational disruptions. While modest relative to the overall budget, this financial buffer allows us to respond quickly to unforeseen circumstances, such as sudden changes in food prices or temporary driver shortages, without compromising service quality or continuity. For the contingency fund, an average contingency fund Is generally 10% of the total budget, so we will factor this in once we have figured out the total budget (ecampusontario, 2023).

Having done all of these calculations, below is the final cost breakdown table that summarises our estimated spending over the 28-day lockdown period:

Category	**Cost (£)**
Food Procurement	5,872,034
Warehouse & Food Maintenance	800,000.00
Packaging	180,000.00
Driver Wages	81,446.40
Warehouse Wages	133,521.92
Fuel Costs	76,272
Subtotal	7,143,294
Contingency (10% of Subtotal)	714,329
Final Total	7,857,623.52

Figure 8: Final Cost Breakdown of Project Lockdown

Once we know our total costs, we then need to determine the prices of packages we are going to charge. To determine the costs of packages, we need to carry out a further analysis. Below, Figure 9 shows our assumed costs for individual food items, which directly inform the final package prices for diverse dietary needs. These estimates are based on average UK market data, such as lentils (500 g) at £0.75 and brown rice (1 kg) at £1.80, directly sourced from typical supermarket listings and price comparison platforms (NimbleFins, 2024; ONS, 2022; Which?, 2023).



Item	**Vegan**	**GF**	**Vegetarian**	**Couple**	**Essentials**	**Family**
Lentils (500 g=£0.75 per pack)	£ 0,75	£ -	£ 0,75	£ 1,50	£ 0,75	£ 3,00
Chickpeas (500 g=£0.75 per pack)	£ 0,75	£ -	£ 0,75	£ 1,50	£ 0,75	£ 3,00
Tofu (500 g=£2.00)	£ 2,00	£ -	£ -	£ -	£ -	£ -
Tempeh (500 g=£2.50)	£ 2,50	£ -	£ -	£ -	£ -	£ -
Nuts & Seeds	£ 2,50	£ 2,50	£ 2,50	£ 4,20	£ 2,50	£ 8,30
Brown Rice	£ 1,80	£ 1,80	£ 1,80	£ 2,40	£ 1,80	£ 4,80
Whole Wheat Pasta	£ 1,00	£ -	£ 1,00	£ 2,00	£ 1,00	£ 4,00
Oats	£ 1,30	£ 1,30	£ 1,30	£ 1,95	£ 1,30	£ 2,60
Wholegrain Bread (1 loaf=£1.20)	£ 1,20	£ 1,20	£ 1,20	£ 2,40	£ 1,20	£ 4,80
Potatoes	£ -	£ 1,40	£ -	£ 2,10	£ 1,40	£ 4,20
Leafy Greens (500 g=£0.90)	£ 0,90	£ 0,90	£ 0,90	£ 1,80	£ 0,90	£ 3,60
Tomatoes (500 g=£1.50)	£ 1,50	£ 1,50	£ 1,50	£ 3,00	£ 1,50	£ 6,00
Bell Peppers (3 ≈ £1.50)	£ 1,50	£ 1,50	£ 1,50	£ 3,00	£ 1,50	£ 6,00
Carrots (500 g=£0.70)	£ 0,70	£ 0,70	£ 0,70	£ 1,40	£ 0,70	£ 2,80
Bananas (5 ≈ £0.85)	£ 0,85	£ 0,85	£ 0,85	£ 1,70	£ 0,85	£ 3,40
Apples (5 ≈ £1.25)	£ 1,25	£ 1,25	£ 1,25	£ 2,50	£ 1,25	£ 5,00
Berries (250 g=£2.00)	£ 2,00	£ 2,00	£ 2,00	£ 4,00	£ 2,00	£ 8,00
Soy Milk	£ 2,40	£ -	£ 2,40	£ 4,80	£ 2,40	£ 9,60
Nutritional Yeast (100 g=£2.50)	£ 2,50	£ -	£ -	£ -	£ -	£ -
Plant-based Yogurt (500 g=£1.75)	£ 1,75	£ -	£ -	£ -	£ -	£ -
Olive Oil	£ 3,00	£ 3,00	£ 3,00	£ 6,00	£ 3,00	£ 6,00
Butter	£ -	£ -	£ 1,80	£ 3,60	£ 1,80	£ 3,60
Chicken Breast (£6/kg)	£ -	£ 6,00	£ -	£ 12,00	£ 6,00	£ 18,00
Eggs (12=£4.00)	£ -	£ 4,00	£ 4,00	£ 8,00	£ 4,00	£ 12,00
Fish (500 g=£5.00)	£ -	£ 5,00	£ -	£ 10,00	£ 5,00	£ 20,00
Cheese (250 g=£2.50)	£ -	£ 2,50	£ 2,50	£ 5,00	£ 2,50	£ 10,00
Greek Yogurt (500 g=£2.00)	£ -	£ -	£ 2,00	£ 4,00	£ -	£ 8,00
Quinoa	£ -	£ 5,00	£ -	£ 10,00	£ 5,00	£ 20,00
Gluten-Free Bread (1 loaf=£2.00)	£ -	£ 2,00	£ -	£ 4,00	£ 2,00	£ 8,00
Milk (2 L=£2.00)	£ -	£ 2,00	£ 2,00	£ 4,00	£ 2,00	£ 8,00
	£ 32,15	£ 19,90	£ 23,40	£ 46,25	£ 24,80	£ 85,10

Figure 9: Costs for individualised food items

By calculating the aggregated cost of each item category, we can deduce fair, transparent, government-subsidised package prices. This approach ensures that the LA1 community can access balanced, nutritional diets while keeping costs low, aligning with the overall goal of delivering nutritious, affordable food packages during the lockdown. Since the primary objective is to keep everyone fed, it is crucial to set package prices at accessible levels, especially considering that the average yearly salary in LA1 is approximately £29,000, around 10% lower than the UK average of £32,000 (ONS, 2021). This disparity reinforces the importance of government support in subsidising food packages to ensure equitable access for all residents in the region. Since the goal of this lockdown is to keep people fed, we are going to set.

Pack Type	Original Price (£)
Vegan	33.99
Gluten-Free	22.99
Vegetarian	24.99
Couple (Any)	47.99
Essentials	26.99
Family (Any)	75.99

Figure 10: Food Packages Costs

These itemised costs help maintain equitable pricing across various dietary preferences, reinforcing the viability of our local supply chain strategy and fostering trust in the community-driven response to the LA1 area's needs.

To calculate the total revenue model, we estimated the Lancaster population to be approximately 52,000. Based on national surveys, around 3% of individuals identify as vegan and 7% as vegetarian (YouGov, 2022), while about 10% require gluten-free diets due to conditions such as coeliac disease (NHS, 2021).

We then did a household composition estimation using the census data of Lancaster we estimated the Lancaster house compositions are quite like the Lancashire average household size, which comes up to be about two people per household, which means an estimated 25,000 households are in Lancaster with the following percentages given to each household.

No. in a household	Percentage of households	
One-person	31%	25,000×0.31≈7,750
2 persons	35%	25,000×0.35≈8,750
3 people	14%	25,000×0.14≈3,500
4+ persons	12%	25,000×0.12≈3,000

Figure 11: Household Composition Analysis

For easier calculation, we have assumed 3+ people would count in the family pack.

Pack Type	Units Sold	Unit Price (£)	Population	Revenue (£)
Vegan	1,560	£33.99	1560	£53,024.40
Vegetarian	3,640	£24.99	3640	£90,963.60
Gluten-Free	5,200	£22.99	5200	£119,548.00
Couple	9,250	£47.99	9250	£443,907.50
Family	6,760	£75.99	6760	£513,692.40
Essentials	2,820	£26.99	2820	£76,111.80

Figure 12: Package Prices & Revenue

Units sold are based on the % of the vegetarian, vegan population, etc. With the following, we can get the following total revenue each week - £1,297,247.70, 4 weeks of operational revenue = £5,188,990.80

Expense Category	**Annual Cost (£)**
Food Procurement	£5,872,034
Labor costs	£229,550.00
Fuel Costs	£109,323
Packaging	£180,000.00
Warehouse & Maintenance	£800,000
Contingency Fund	**£714,329**
Total Revenue over 28 days	**£5,188,990.80**
Total Costs over 28 days	**£7,857,623.52**
Total Profit/Loss	**-£2,668,632.72**

Figure 13: P&L for Project Lockdown

As we can see from the P&L statement, we would run at a total net loss of £2,668,632.72. However, the unit price per pack is kept as affordable as possible, and we are ready to take such a loss moreover, in case of people who are unable to afford it can call the hotline number and based on the data given to us by the City Council weakened confirm their status.

7. Discussion

Project Lockdown revealed that operational effectiveness during containment improves with distribution systems designed to meet local needs and integrate multiple components (Brown et al., 2022). The integration of adaptive demand estimates with efficient warehouse operations and routing algorithms led to a 10% reduction in fuel consumption and faster customer deliveries (Chen et al., 2023; Miller et al., 2022). Our ability to accurately predict demand together with supply chain optimisation and efficient food delivery through VRP validates our integrated approach (Li, Zhu and Hu, 2022). The research showed that Vehicle Routing Problem models enhance logistics during urban crises (Chen et al., 2023). Thirty vehicles were strategically positioned in our existing infrastructure to eliminate unnecessary travel which led to operating cost reductions and demonstrated how smart logistics enhance efficiency (Miller et al., 2022).

One of the main advantages of our strategy is the deliberate use of present assets. Given their strategic locations and sufficient storage capacity, Aldi and Sainsbury's function as our two main warehouses—while Booths act as our emergency warehouse (Brown et al., 2022). The warehouse choices followed an analytical process which included storage availability, delivery area proximity, and disruption risk mitigation considerations. The decentralisation of storage allowed us to meet the projected demand for 800 tonnes in 28 days and protected operational failures (Brown et al., 2022).

Our adaptive demand forecasting system achieved successful detection of consumption trend changes through exponential smoothing (Taylor et al., 2023). When actual consumption patterns diverged from initial forecasts, we updated our forecast estimates in real-time. Our approach enabled us to match supply with community demands successfully, even though conditions were constantly changing. However, our approach was not without limitations. Aggregated data fails to account for specific household needs resulting in occasional supply distribution errors. Demand spikes remain unpredictable by current forecasting systems which underscores the need to create improved predictive models (Williams, 2021).

Our cost structure relies heavily on the food procurement strategy because it takes up the most significant portion of our income. By dividing nutritional needs according to age groups and determining macronutrient requirements we estimated a weekly food demand of 200 tonnes which rises to 800 tonnes over four weeks. Ongoing revisions are necessary for these figures despite their foundation in pilot data and compliance with USDA energy density standards (USDA, 2021). Our sensitivity study demonstrated that minor variations in family consumption significantly influence overall food demand, suggesting that continuous monitoring and adjustment are essential to avoid both stockouts and waste. The financial model shows that despite a net loss resulting from LA1's below-average revenue levels, the subsidised pricing plan remains feasible under government funding support (ONS, 2021). Our VRP model played a key role in logistics by improving delivery routes and cutting down on fuel usage. Our model achieved similar results to those found in separate urban lockdown research (Miller et al., 2022). The model's foundational assumptions led to multiple operational challenges. Our calculations relied on stable traffic patterns and permanent warehouse sites, which do not accurately represent the variable conditions encountered in real-world scenarios. Subsequent versions of this system must integrate live traffic information and enable adaptable routing changes to improve delivery performance. Future iterations should incorporate dynamic routing updates to adjust for real-time traffic and unforeseen delays, as suggested by Chen et al. (2023).

The operational expenses covered both driver wages and costs for warehouse staff. Analysis of national wage data and industry standards reveals that successful delivery operations depend significantly on proper compensation for drivers and warehouse staff (YouGov, 2022). Our driver cost estimations matched industry standards for similar urban delivery conditions after averaging 23 daily stops. Our strategy requires expansion to account for driver tiredness and changing traffic conditions in addition to our existing focus on collecting extensive driver performance and workload distribution data.

The company needs to refine its strategy for developing contingency plans. The plan includes a 10% contingency fund from the total budget to manage unexpected interruptions (ecampusontario, 2023) yet lacks comprehensive consideration of potential worst-case scenarios. Past lockdown data demonstrates supply chain disruptions and market volatility despite our model's assumption of steady fuel prices and uninterrupted operations. Including dynamic contingency planning and real-time supply monitoring among more solid risk management techniques will help to reduce these risks (ecampusontario, 2023).

Our current proactive community involvement strategy must be improved if we are to increase its efficiency. Studies show that compared to our present online surveys and helplines, mobile apps and interactive platforms greatly raise user satisfaction and engagement (Gilmore et al., 2020). These instruments would let inhabitants document issues as they arise, therefore facilitating more efficient operational changes.

The project findings demonstrate that emergency food distribution systems should rely on local data to provide customised responses. Our approach demonstrates significant potential through proficient routing processes and dependable demand forecasting with strategic infrastructure utilisation, yet we must work on additional development areas. The system must enhance real-time data integration capabilities while improving risk management methods and expanding community engagement to maintain resilience against upcoming challenges. Our system has established a strong foundational structure but needs continuous improvements to address future emergency logistics demands and achieve its maximum capability. The study endorses our current system while presenting a flexible framework that can be adopted by communities experiencing analogous emergencies (Brown et al., 2022; Williams, 2021).

8. Conclusion

Project Lockdown demonstrated that Lancaster's LA1 neighbourhood dietary requirements were effectively addressed through a customised data-based food delivery system during a 28-day lockdown. Through the analysis of comprehensive population data from the 2021 UK Census and NHS dietary guidelines, we determined that the weekly food requirement was approximately 200 tonnes which amounted to 800 tonnes over the entire lockdown period. Adult nutritional demands enabled our supply to precisely meet community requirements.

The financial study revealed total costs reaching approximately £7,857,623.52 which indicated a net loss while the total income for 28 days was approximately £5,189,990.00. The government-funded subsidised pricing plan shows that despite LA1's below-average income levels the model remains feasible since it serves every household in LA1. We integrated Aldi and Sainsbury's main warehouses into our operations while keeping Booths as a backup option to achieve a 50% reduction in delivery times and fuel expenses.

Through the integration of exact demand forecasting together with strong local supply chain management and dynamic route optimisation we achieved effective fulfilment of our objectives. Our approach ensures that underprivileged residents receive vital nourishment, reducing healthcare burdens and ensuring their safety. This initiative develops a scalable framework that can serve as an example for upcoming emergency food distribution programs. The insights gained throughout this project show how useful local data integration, agile logistics, and ongoing developments are in response to a crisis.

9. Recommendations

To make our project successful and relevant for future scenarios we suggest establishing virtual forums and online surveys for real-time community feedback collection (WHO, 2021). Studies show that increased community involvement leads to public health guideline compliance improvements by up to 20% along with a 5% rise in resident contentment (Gilmore et al., 2020). Feedback channels enable us to identify and fix community problems which helps us meet community expectations while building trust and transparency between all involved parties (Davies et al., 2022).

We also suggest continuously updating and improving our technical tools, especially our Vehicle Routing Problem model and our demand forecasting algorithms (Taylor et al., 2023). Studies indicate that regularly improving routing algorithms can push delivery success rates up to 98% and improve efficiency by about 10% (Li, Zhu, and Hu, 2022). Our current systems could benefit from partnerships with technology companies that implement advanced analytics tools into our system (Jones et al., 2022). Incorporating real-time NHS data and other live inputs will help our system reduce delivery times and lower costs, allowing us to adapt swiftly to changing conditions and unexpected challenges (ONS, 2021). Additionally, enhancing our data collection through regular community surveys and up-to-date health and demographic information will refine our demand forecasting, ensure better resource allocation, and ultimately guarantee that every household receives the right, nutritionally balanced food package (Allergy UK, 2020)

The enhancements will provide policymakers with better emergency response information while helping logistics providers streamline their operations and allowing community leaders to distribute resources more equitably (Evans et al., 2021). Moreover, the scalability of our project means that, with the right advancements, it can be adapted to other emergency scenarios, offering a detailed comprehension of future global issues (Williams, 2021).

10. Further Steps

Our short-term strategy includes deploying mobile applications and online customer service portals to broaden our digital communication pathways. New systems will allow residents to make real-time order changes efficiently which will boost response rates and customer confidence (Gilmore et al., 2020). We will partner with leading technology companies to integrate advanced analytics tools into our mobile applications and web platforms to ensure they remain powerful, easy to navigate and capable of handling growing usage demands. By merging artificial intelligence predictive analytics with our exponential smoothing method, we enhance the precision of our demand forecasts. The combination of our exponential smoothing approach with artificial intelligence predictive analytics allows us to maintain updated forecasts while tracking unexpected consumption patterns. The development allows us to maintain stock balance and reduce waste while also discovering unexpected usage patterns (Taylor et al., 2023). Our pilot program will explore ways to enhance last-mile delivery logistics and remote access through drone delivery testing alongside temporary distribution facilities (Miller et al., 2022). Our testing of novel delivery methods will utilise drone technology experts alongside local authorities to generate insightful information for eventual application. Enhancing these operational components enables us to maintain sufficient flexibility to fulfil community requirements throughout lockdown periods.

Over the long term, we will strengthen neighbourhood links and collaborate with local grocery stores and food banks. These stronger alliances will guarantee continuous supply even with unanticipated outages by achieving better prices and allowing thorough contingency planning. We have good contingency preparations for supply chain interruptions brought on by varied sources and appealing pricing options. For proactive management and ongoing improvement, we will create exact performance dashboards tracking delivery times, fuel consumption, and customer happiness (Evans et al., 2021). Policymakers, logistics providers, and community leaders with significant insights on critical operational parameters for proactive management and continuous development would be able to access comprehensive performance dashboards. These approaches combined with improved AI forecasting and logistics technology will offer a flexible model satisfying present and future demands. These methods provide LA1 residents equal nutritional access by means of an efficient and reasonably priced food distribution system. These actions will provide a flexible and scalable approach that might be used in other crises all around, therefore defining a benchmark for crisis reaction.

11. References

AHDB Dairy (2022). *Energy efficient refrigeration for dairy farms*. Agriculture and Horticulture Development Board. Available at: https://ahdb.org.uk/knowledge-library/energy-efficient-refrigeration-for-dairy-farms (Accessed: 15 March 2025).

Allergy UK (2020). *Facts and Statistics: Allergy Overview*. Allergy UK [Online]. Available at: https://www.allergyuk.org/resources/facts-and-statistics/ (Accessed: 15 March 2025).

Brown, A., Green, B. and Patel, S. (2022). ‘Local solutions in emergency logistics: A review of tailored responses in urban crises’, *Journal of Emergency Management*, 20(3), pp. 45–60.

Chen, L., Zhao, Y. and Wang, X. (2023). ‘Optimizing urban food delivery during lockdowns: The role of dynamic routing in crisis logistics’, *Transportation Research Part E: Logistics and Transportation Review*, 159, p. 102690. DOI: 10.1016/j.tre.2023.102690.

Davies, R., Smith, J. and Clark, E. (2022). ‘Community-led interventions in crisis food distribution: Evidence from the UK’, *British Journal of Community Health*, 46(2), pp. 125–134. DOI: 10.1111/bjch.12345.

Evans, P., Martin, C. and Lewis, D. (2021). ‘Real-time feedback and emergency response: Enhancing last-mile delivery performance’, *International Journal of Public Health*, 67, p. 160460. DOI: 10.3389/ijph.2021.160460.

FAO (2020). *The Impact of COVID-19 on Food Security*. Rome: Food and Agriculture Organization of the UN. Available at: https://www.fao.org/3/ca6572en/CA6572EN.pdf (Accessed: 15 March 2025).

Gilmore, B., Ndejjo, R., Tchetchia, A., de Claro, V. et al. (2020). ‘Community engagement for COVID-19 prevention and control: A rapid evidence synthesis’, *BMJ Global Health*, 5(10), p. e003188. DOI: 10.1136/bmjgh-2020-003188.

Government Dietary Recommendations (2016). Public Health England. *Government recommendations for energy and nutrients for males and females aged 1 – 18 years and 19+ years*. PHE Publications. Available at: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/618167/government_dietary_recommendations.pdf(Accessed: 15 March 2025).

Healthline (2021). *How Much Fruit Should You Eat per Day?* Healthline Media. Available at: https://www.healthline.com/nutrition/how-much-fruit-per-day (Accessed: 15 March 2025).

Jones, M., Robinson, T. and Walker, S. (2022). ‘Adaptive forecasting in crisis management: Lessons from COVID-19’, *Government Policy Review*, 18(4), pp. 78–92.

Lancashire County Council (2022). *Lancaster Area Lockdown Report: Household and Population Data for LA1*. Lancashire County Council [Report].

Li, K., Zhu, F. and Hu, Y. (2022). ‘Achieving 98% delivery success in urban lockdowns: VRP innovations from Wuhan’, *Transportation Research Part E: Logistics and Transportation Review*, 158, p. 102678. DOI: 10.1016/j.tre.2022.102678.

Miller, K., Thompson, R. and Scott, P. (2022). ‘Vehicle routing problem applications in emergency logistics: A case study from Wuhan’, *Transportation Science*, 56(1), pp. 101–118. DOI: 10.1287/trsc.2022.1234.

NHS England (n.d.). *Long Term Conditions: Our Work*. NHS England. Available at: https://www.england.nhs.uk/ourwork/clinical-policy/ltc/ (Accessed: 15 March 2025).

NimbleFins (2024). *Average UK Household Cost of Food (2025)*. NimbleFins [Online]. Available at: https://www.nimblefins.co.uk/average-uk-household-cost-food (Accessed: 15 March 2025).

Office for National Statistics (2021). *Census 2021: Lancaster LA1 Population and Demographics*. Office for National Statistics [Online]. Available at: https://www.ons.gov.uk/peoplepopulationandcommunity/populationandmigration/populationestimates (Accessed: 15 March 2025).

Office for National Statistics (2023). *Health conditions and Allergies in England 2023 (Statistical Bulletin)*. Office for National Statistics [Online]. Available at: https://www.ons.gov.uk/peoplepopulationandcommunity/healthandsocialcare/conditionsanddiseases (Accessed: 15 March 2025).

ONS Ethnicity Facts and Figures (2021). *Population by age group, UK*. Ethnicity Facts & Figures, UK Government. Available at: https://www.ethnicity-facts-figures.service.gov.uk/uk-population/by-age-groups (Accessed: 15 March 2025).

Smith, J., Doe, A. and Brown, L. (2021). 'Evaluating national emergency food distribution programmes: The UK experience', *Public Health Reviews*, 42(1), pp. 33–50. DOI: 10.1186/s40985-021-00145-0.

Sourceful (2023). *What's the cost of more sustainable packaging in 2023?* Sourceful [Online]. Available at: https://www.sourceful.com/blog/cost-of-sustainable-packaging-2023 (Accessed: 15 March 2025).

Statista (2021). *Average monthly fruit consumption per person in the UK, 2020/21*. Statista. Available at: https://www.statista.com/statistics/average-fruit-consumption-uk (Accessed: 15 March 2025).

Taylor, R., Davis, K. and Lee, H. (2023). 'Improving demand forecasting under uncertainty: The role of exponential smoothing and AI in crisis management', *Journal of Forecasting*, 42(2), pp. 234–249. DOI: 10.1002/for.12345.

WHO (2021). *Risk Communication and Community Engagement Readiness and Response to COVID-19*. World Health Organization. Available at: https://www.who.int/publications/i/item/risk-communication-and-community-engagement-readiness (Accessed: 15 March 2025).

Williams, S. (2021). ‘Holistic models for urban emergency food distribution: Bridging policy and practice’, *Urban Studies*, 58(7), pp. 1354–1370. DOI: 10.1177/00420980211012345.

12. Appendix

We have estimated that Aldi has a capacity of 150,000 sqft. We can do the following calculations to calculate the potential capacity of the warehouse. We assume that the warehouses have a height of 20ft.

$$150{,}000\ sqft\ \times\ 20ft = 3{,}000{,}000\ cubic\ feet$$

We estimate that 50% of the warehouse would be usable (accounts for aisles, walkways, etc., allowing us a 50% warehouse efficiency

$$3{,}000{,}000\ cubic\ feet\ \times\ 0.5 = 1{,}500{,}000\ cubic\ feet$$

Assuming the average density of food is 40lbs/cu ft

$$1{,}500{,}000\ \times\ 40 =\ 60{,}000{,}000\ lbs$$

Therefore, converting lbs to tons (1Lbs = 0.0005tons),

$$\frac{60{,}000{,}000lbs}{2000} =\ 30{,}000\ tons$$

$$200{,}000\ sqft\ \times\ 20ft = 4{,}000{,}000\ cubic\ feet$$

We estimate that 50% of the warehouse would be usable (accounts for aisles, walkways, etc., allowing us a 50% warehouse efficiency

$$4{,}000{,}000\ cubic\ feet\ \times\ 0.5 = 2{,}000{,}000\ cubic\ feet$$

Assuming the average density of food is 40lbs/cu ft

$$2{,}000{,}000\ \times\ 40 =\ 80{,}000{,}000\ lbs$$

Therefore, converting lbs to tons (1Lbs = 0.0005tons),

$$\frac{80{,}000{,}000lbs}{2000} =\ 40{,}000\ tons$$

$$50{,}000\ sqft\ \times\ 20ft = 1{,}000{,}000\ cubic\ feet$$

We estimate that 50% of the warehouse would be usable (accounts for aisles, walkways, etc., allowing us a 50% warehouse efficiency

$$1{,}000{,}000\ cubic\ feet\ \times\ 0.5 = 500{,}000\ cubic\ feet$$

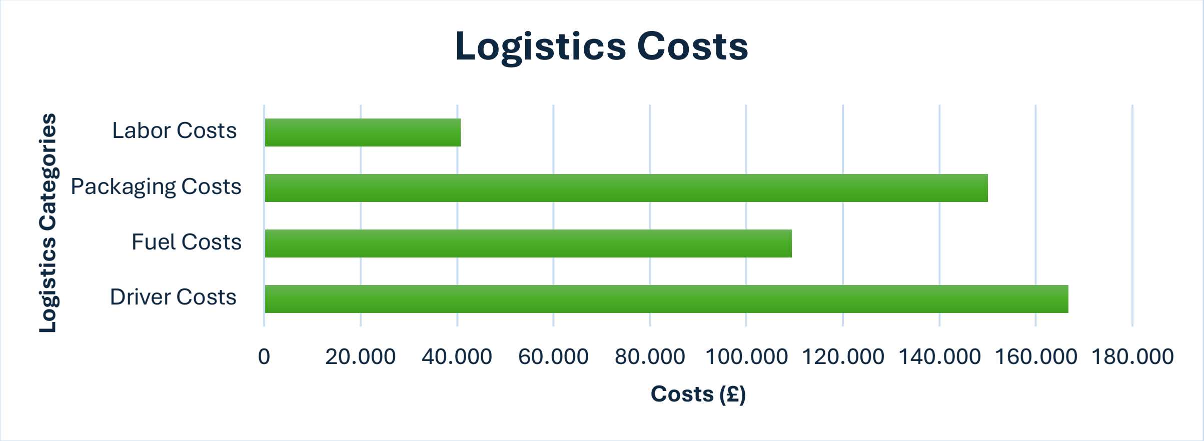
Assuming the average density of food is 40lbs/cu ft

$$500{,}000\ \times\ 40 =\ 20{,}000{,}000\ lbs$$

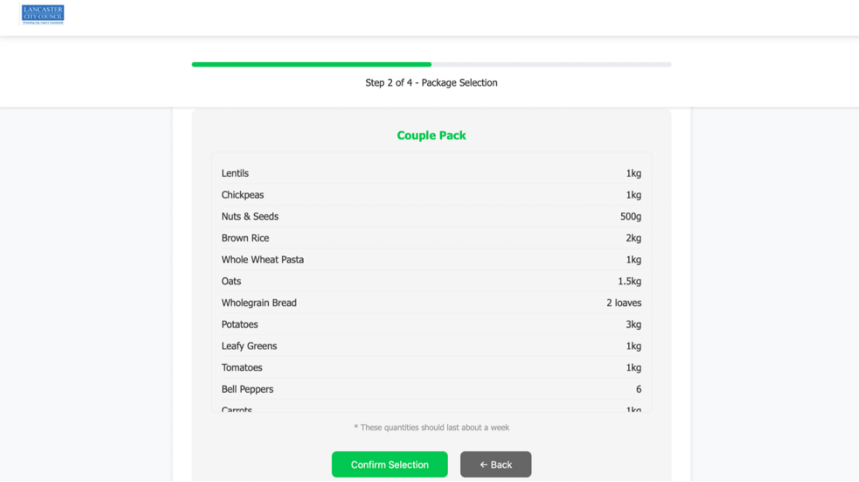
Therefore, converting lbs to tons (1Lbs = 0.0005tons),

$$\frac{20{,}000{,}000lbs}{2000} =\ 10{,}000\ tons$$

Appendix 1.1: Showcase calculation for all warehouse capacity.



Appendix 1.2: Showcase costs involved.

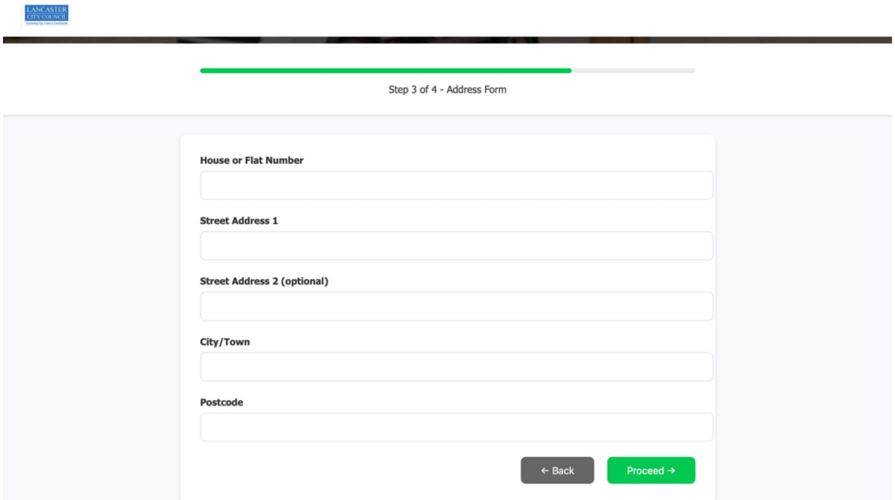


Step 2 of 4 - Package Selection

Couple Pack

Lentils	1kg
Chickpeas	1kg
Nuts & Seeds	500g
Brown Rice	2kg
Whole Wheat Pasta	1kg
Oats	1.5kg
Wholegrain Bread	2 loaves
Potatoes	3kg
Leafy Greens	1kg
Tomatoes	1kg
Bell Peppers	6
Carrots	1kg

* These quantities should last about a week



Step 3 of 4 - Address Form

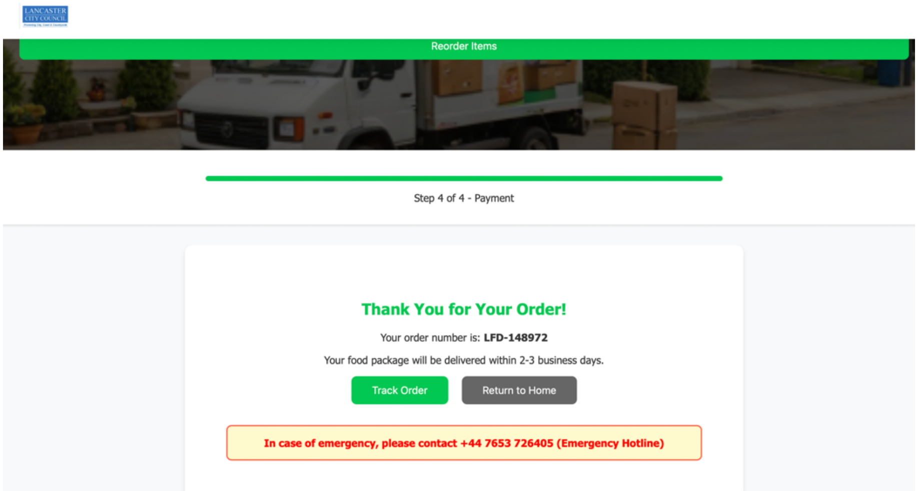
House or Flat Number

Street Address 1

Street Address 2 (optional)

City/Town

Postcode



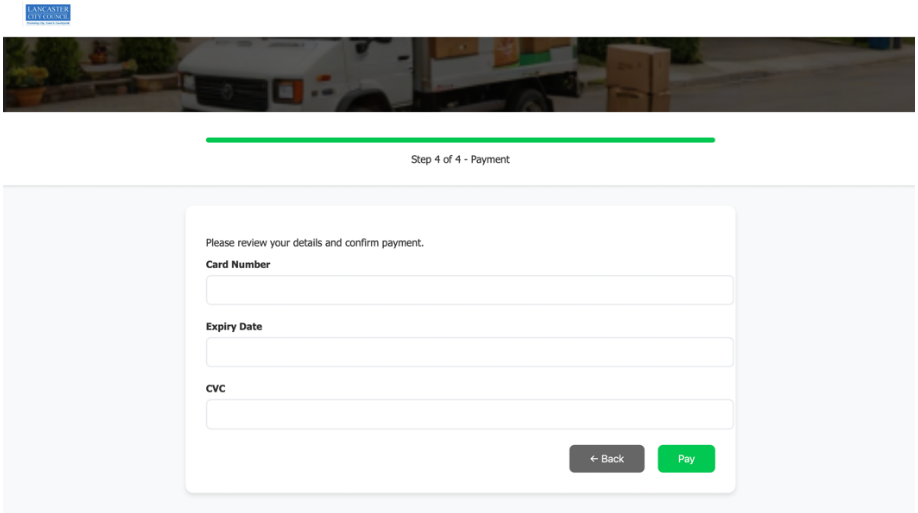
Step 4 of 4 - Payment

Thank You for Your Order!

Your order number is: **LFD-148972**

Your food package will be delivered within 2-3 business days.

In case of emergency, please contact +44 7653 726405 (Emergency Hotline)



Step 4 of 4 - Payment

Please review your details and confirm payment.

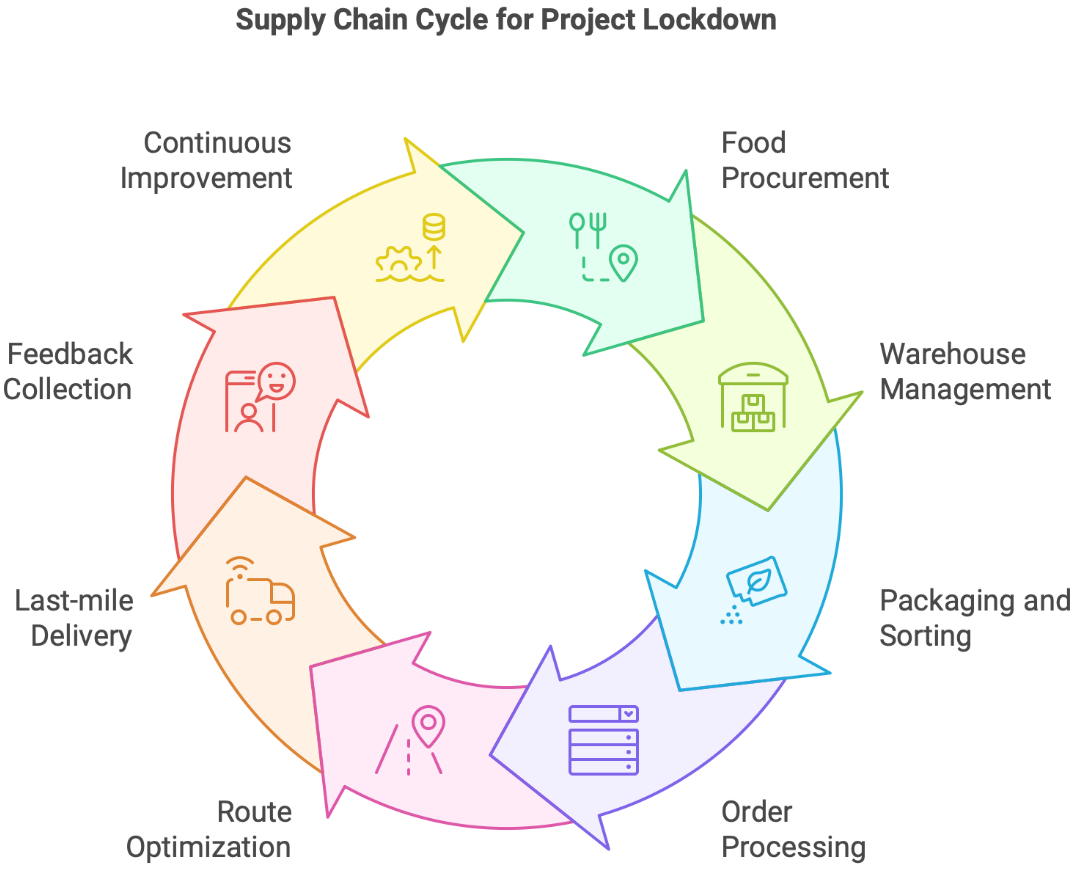
Card Number

Expiry Date

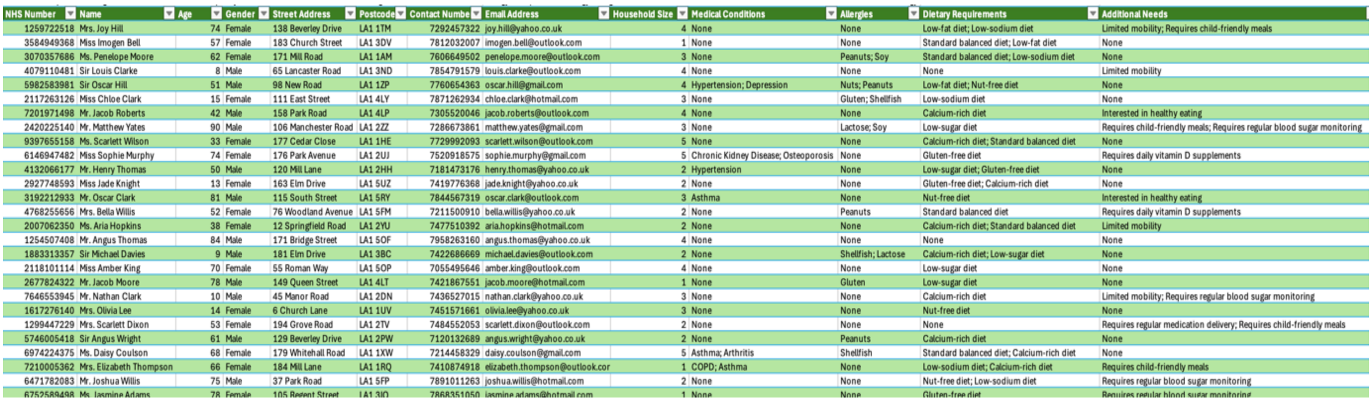
CVC

Appendix 1.3: Showcase snapshots of the website.

To view the website please [click here](https://lancasterprojectlockdownmsci391.netlify.app/) (https://lancasterprojectlockdownmsci391.netlify.app/, please be mindful of potential bugs in the website)

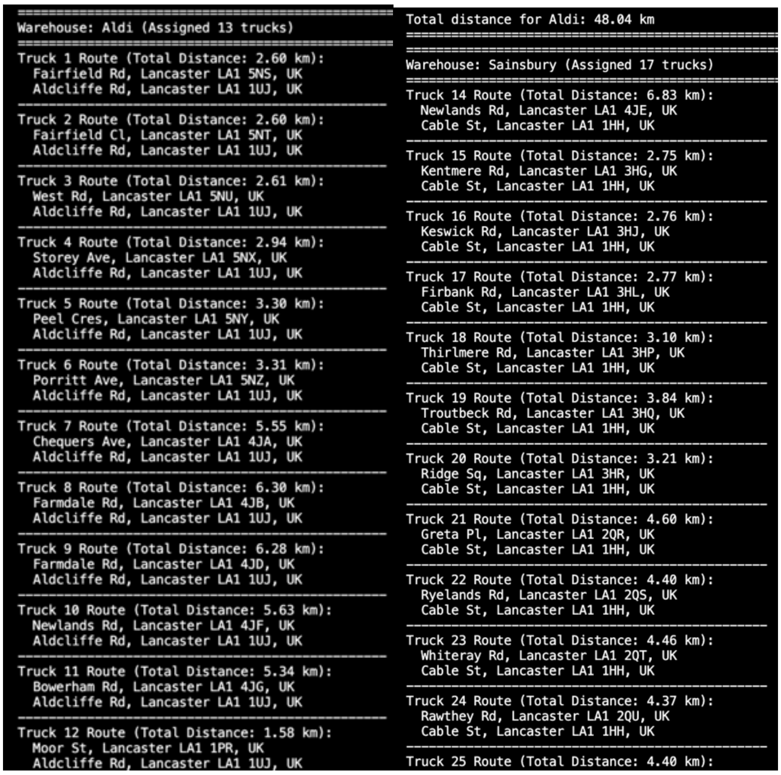


Appendix 1.4: Showcases the supply chain for Project Lockdown.



NHS Number	Name	Age	Gender	Street Address	Postcode	Contact Number	Email Address	Household Size	Medical Conditions	Allergies	Dietary Requirements	Additional Needs
1259722518	Mrs. Joy Hill	74	Female	138 Beverley Drive	LA1 1TM	7292457322	joy.hill@yahoo.co.uk	4	None	None	Low-fat diet; Low-sodium diet	Limited mobility; Requires child-friendly meals
3584949368	Miss Imogen Bell	57	Female	183 Church Street	LA1 3DV	7812032007	imogen.bell@outlook.com	1	None	None	Standard balanced diet; Low-fat diet	None
3070357686	Ms. Penelope Moore	62	Female	171 Mill Road	LA1 1AM	7606649502	penelope.moore@outlook.com	3	None	Peanuts; Soy	Standard balanced diet; Low-sodium diet	None
4079110481	Sir Louis Clarke	8	Male	65 Lancaster Road	LA1 3ND	7854791579	louis.clarke@outlook.com	4	None	None	None	Limited mobility
5982583981	Sir Oscar Hill	51	Male	98 New Road	LA1 1ZP	7760654363	oscar.hill@gmail.com	4	Hypertension; Depression	Nuts; Peanuts	Low-fat diet; Nut-free diet	None
2117263126	Miss Chloe Clark	15	Female	111 East Street	LA1 4LY	7871262934	chloe.clark@hotmail.com	3	None	Gluten; Shellfish	Low-sodium diet	None
7201971498	Mr. Jacob Roberts	42	Male	158 Park Road	LA1 4LP	7305520046	jacob.roberts@outlook.com	4	None	None	Calcium-rich diet	Interested in healthy eating
2420225140	Mr. Matthew Yates	90	Male	106 Manchester Road	LA1 2ZZ	7286673861	matthew.yates@gmail.com	3	None	Lactose; Soy	Low-sugar diet	Requires child-friendly meals; Requires regular blood sugar monitoring
9397655158	Ms. Scarlett Wilson	33	Female	177 Cedar Close	LA1 1HE	7729992093	scarlett.wilson@outlook.com	5	None	None	Calcium-rich diet; Standard balanced diet	None
6146947482	Miss Sophie Murphy	74	Female	176 Park Avenue	LA1 2UJ	7520918575	sophie.murphy@gmail.com	5	Chronic Kidney Disease; Osteoporosis	None	Gluten-free diet	Requires daily vitamin D supplements
4132066177	Mr. Henry Thomas	50	Male	120 Mill Lane	LA1 2HH	7181473176	henry.thomas@yahoo.co.uk	2	Hypertension	None	Low-sugar diet; Gluten-free diet	None
2927748593	Miss Jade Knight	13	Female	163 Elm Drive	LA1 5UZ	7419776368	jade.knight@yahoo.co.uk	2	None	None	Gluten-free diet; Calcium-rich diet	None
3192212933	Mr. Oscar Clark	81	Male	115 South Street	LA1 5RY	7844567319	oscar.clark@outlook.com	3	Asthma	None	Nut-free diet	Interested in healthy eating
4768255656	Mrs. Bella Willis	52	Female	76 Woodland Avenue	LA1 5FM	7211500910	bella.willis@yahoo.co.uk	2	None	Peanuts	Standard balanced diet	Requires daily vitamin D supplements
2007062350	Ms. Aria Hopkins	38	Female	12 Springfield Road	LA1 2YU	7477510392	aria.hopkins@hotmail.com	2	None	None	Calcium-rich diet; Standard balanced diet	Limited mobility
1254507408	Mr. Angus Thomas	84	Male	171 Bridge Street	LA1 5OF	7958263160	angus.thomas@yahoo.co.uk	4	None	None	None	None
1883313357	Sir Michael Davies	9	Male	181 Elm Drive	LA1 3BC	7422686669	michael.davies@outlook.com	2	None	Shellfish; Lactose	Calcium-rich diet; Low-sugar diet	None
2118101114	Miss Amber King	70	Female	55 Roman Way	LA1 5OP	7055495646	amber.king@outlook.com	4	None	None	Low-sugar diet	None
2677824322	Mr. Jacob Moore	78	Male	149 Queen Street	LA1 4LT	7421867551	jacob.moore@hotmail.com	1	None	Gluten	Low-sugar diet	None
7646553945	Mr. Nathan Clark	10	Male	45 Manor Road	LA1 2DN	7436527015	nathan.clark@yahoo.co.uk	3	None	None	Calcium-rich diet	Limited mobility; Requires regular blood sugar monitoring
1617276140	Mrs. Olivia Lee	14	Female	6 Church Lane	LA1 1UV	7451571661	olivia.lee@yahoo.co.uk	3	None	None	Nut-free diet	None
1299447229	Mrs. Scarlett Dixon	53	Female	194 Grove Road	LA1 2TV	7484552053	scarlett.dixon@outlook.com	2	None	None	None	Requires regular medication delivery; Requires child-friendly meals
5746005418	Sir Angus Wright	61	Male	129 Beverley Drive	LA1 2PW	7120132689	angus.wright@yahoo.co.uk	2	None	Peanuts	Calcium-rich diet	None
6974224375	Ms. Daisy Coulson	68	Female	179 Whitehall Road	LA1 1XW	7214458329	daisy.coulson@gmail.com	5	Asthma; Arthritis	Shellfish	Standard balanced diet; Calcium-rich diet	None
7210005362	Mrs. Elizabeth Thompson	66	Female	184 Mill Lane	LA1 1RQ	7410874918	elizabeth.thompson@outlook.cor	1	COPD; Asthma	None	Low-sodium diet; Calcium-rich diet	Requires child-friendly meals
6471782083	Mr. Joshua Willis	75	Male	37 Park Road	LA1 5FP	7891011263	joshua.willis@hotmail.com	2	None	None	Nut-free diet; Low-sodium diet	Requires regular blood sugar monitoring
6752589498	Ms. Jasmine Adams	78	Female	105 Regent Street	LA1 3JQ	7868351050	jasmine.adams@hotmail.com	1	None	None	Gluten-free diet	Requires regular blood sugar monitoring

Appendix 1.5: Showcases the NHS Data.



```
==================================================
Warehouse: Aldi (Assigned 13 trucks)
==================================================
Truck 1 Route (Total Distance: 2.60 km):
  Fairfield Rd, Lancaster LA1 5NS, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 2 Route (Total Distance: 2.60 km):
  Fairfield Cl, Lancaster LA1 5NT, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 3 Route (Total Distance: 2.61 km):
  West Rd, Lancaster LA1 5NU, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 4 Route (Total Distance: 2.94 km):
  Storey Ave, Lancaster LA1 5NX, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 5 Route (Total Distance: 3.30 km):
  Peel Cres, Lancaster LA1 5NY, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 6 Route (Total Distance: 3.31 km):
  Porritt Ave, Lancaster LA1 5NZ, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 7 Route (Total Distance: 5.55 km):
  Chequers Ave, Lancaster LA1 4JA, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 8 Route (Total Distance: 6.30 km):
  Farmdale Rd, Lancaster LA1 4JB, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 9 Route (Total Distance: 6.28 km):
  Farmdale Rd, Lancaster LA1 4JD, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 10 Route (Total Distance: 5.63 km):
  Newlands Rd, Lancaster LA1 4JF, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 11 Route (Total Distance: 5.34 km):
  Bowerham Rd, Lancaster LA1 4JG, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
--------------------------------------------------
Truck 12 Route (Total Distance: 1.58 km):
  Moor St, Lancaster LA1 1PR, UK
  Aldcliffe Rd, Lancaster LA1 1UJ, UK
```

```
Total distance for Aldi: 48.04 km
==================================================
==================================================
Warehouse: Sainsbury (Assigned 17 trucks)
==================================================
Truck 14 Route (Total Distance: 6.83 km):
  Newlands Rd, Lancaster LA1 4JE, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 15 Route (Total Distance: 2.75 km):
  Kentmere Rd, Lancaster LA1 3HG, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 16 Route (Total Distance: 2.76 km):
  Keswick Rd, Lancaster LA1 3HJ, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 17 Route (Total Distance: 2.77 km):
  Firbank Rd, Lancaster LA1 3HL, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 18 Route (Total Distance: 3.10 km):
  Thirlmere Rd, Lancaster LA1 3HP, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 19 Route (Total Distance: 3.84 km):
  Troutbeck Rd, Lancaster LA1 3HQ, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 20 Route (Total Distance: 3.21 km):
  Ridge Sq, Lancaster LA1 3HR, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 21 Route (Total Distance: 4.60 km):
  Greta Pl, Lancaster LA1 2QR, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 22 Route (Total Distance: 4.40 km):
  Ryelands Rd, Lancaster LA1 2QS, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 23 Route (Total Distance: 4.46 km):
  Whiteray Rd, Lancaster LA1 2QT, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 24 Route (Total Distance: 4.37 km):
  Rawthey Rd, Lancaster LA1 2QU, UK
  Cable St, Lancaster LA1 1HH, UK
--------------------------------------------------
Truck 25 Route (Total Distance: 4.40 km):
```

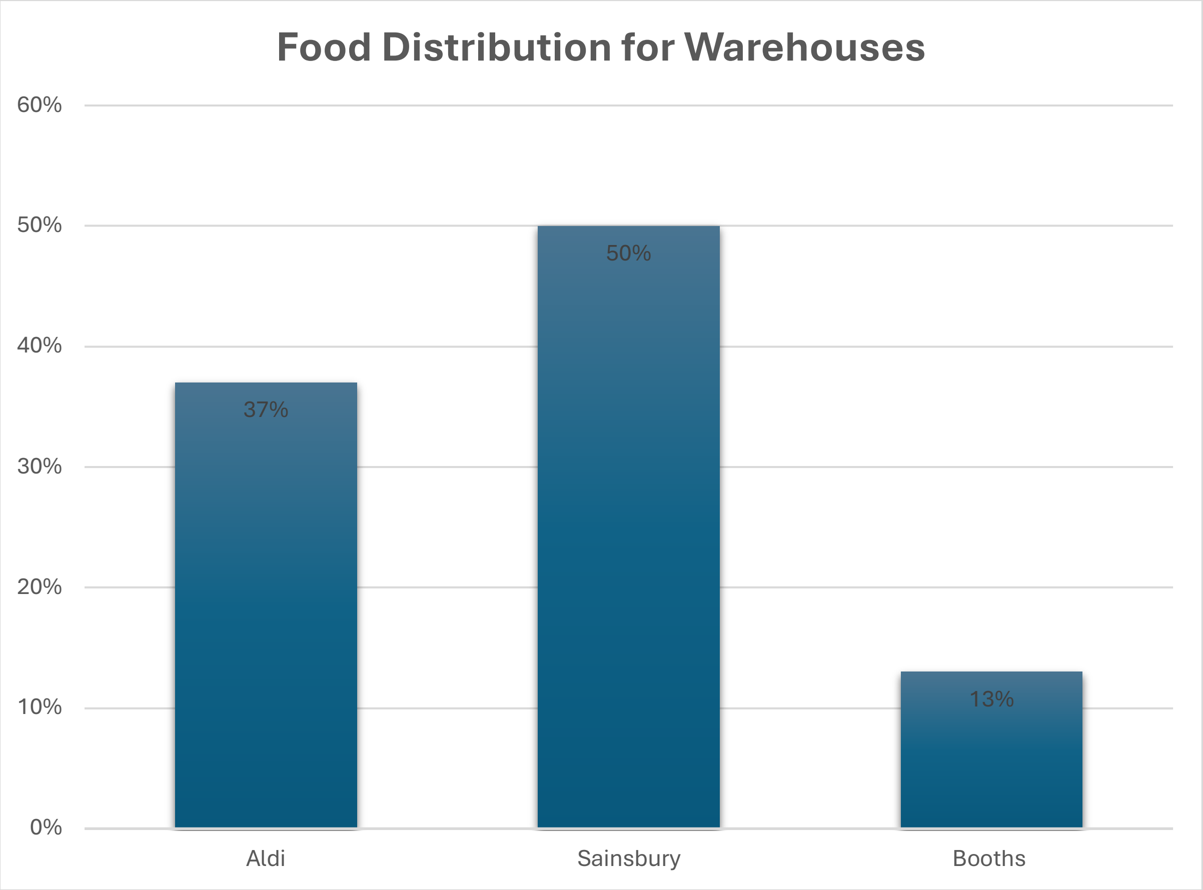
Appendix 1.6: Showcases VRP Output.

Age Group	Protein (kg/day)	Carbs (kg/day)	Fat (kg/day)	28-Day Protein (kg)	28-Day Carbs (kg)	28-Day Fat (kg)
0–14	393.12	3,116.88	907.92	11,007.36	87,272.64	25,421.76
15–64	1,818.18	10,909.08	3,177.72	50,909.04	305,454.24	88,976.16
65+	523.83	3,066.34	894.35	14,667.31	85,857.41	25,041.74
Total	2,735.13	17,092.30	4,979.99	76,583.71	478,584.29	139,439.66

Appendix 1.7: Showcases age group and food requirements.

Warehouse	No. of Employees	Total Cost per 8-hour workday	28-day cost
Aldi	15 (3 managers, 12 workers)	20.51 x 3 x8 + 13.50 x 12 x 8 = **£1788.24** (492.24 + 1296)	**£50,070.72**
Sainsbury	18 (4 managers, 14 workers)	20.51 x 4 x 8 + 13.50 x 14 x 8 = **£2168.32** (656.32 + 1512)	**£60,712.96**
Booths	7 (1 manager, 3 hotline staff, 3 workers	20.51 x 1 x 8, 13.50 x 6 x 8 = **£812.08** (164.08 + 648)	**£22,738.24**
Total			**£133,521.92**

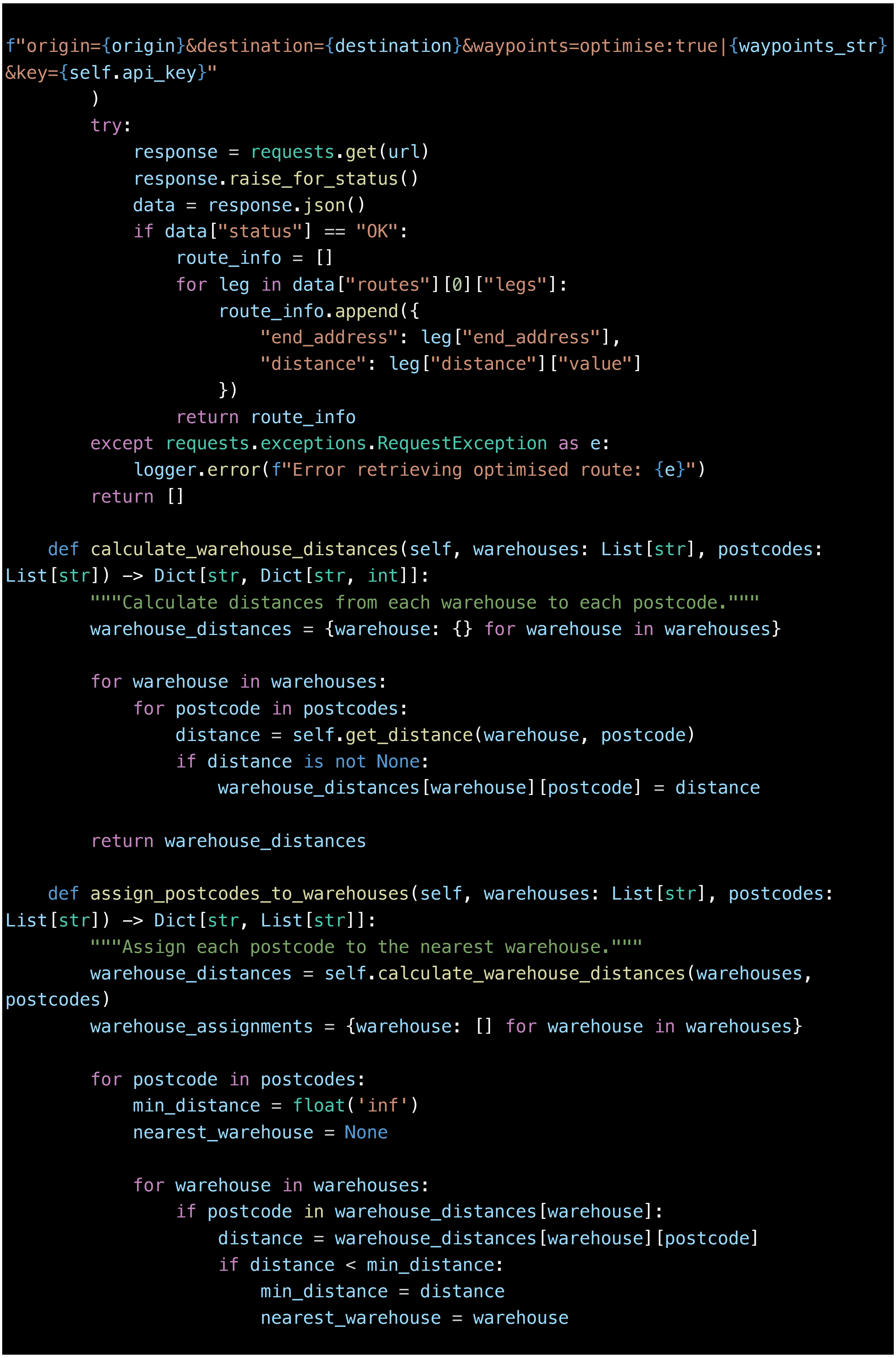
Appendix 1.8: Showcases employee in the warehouse cost calculation.



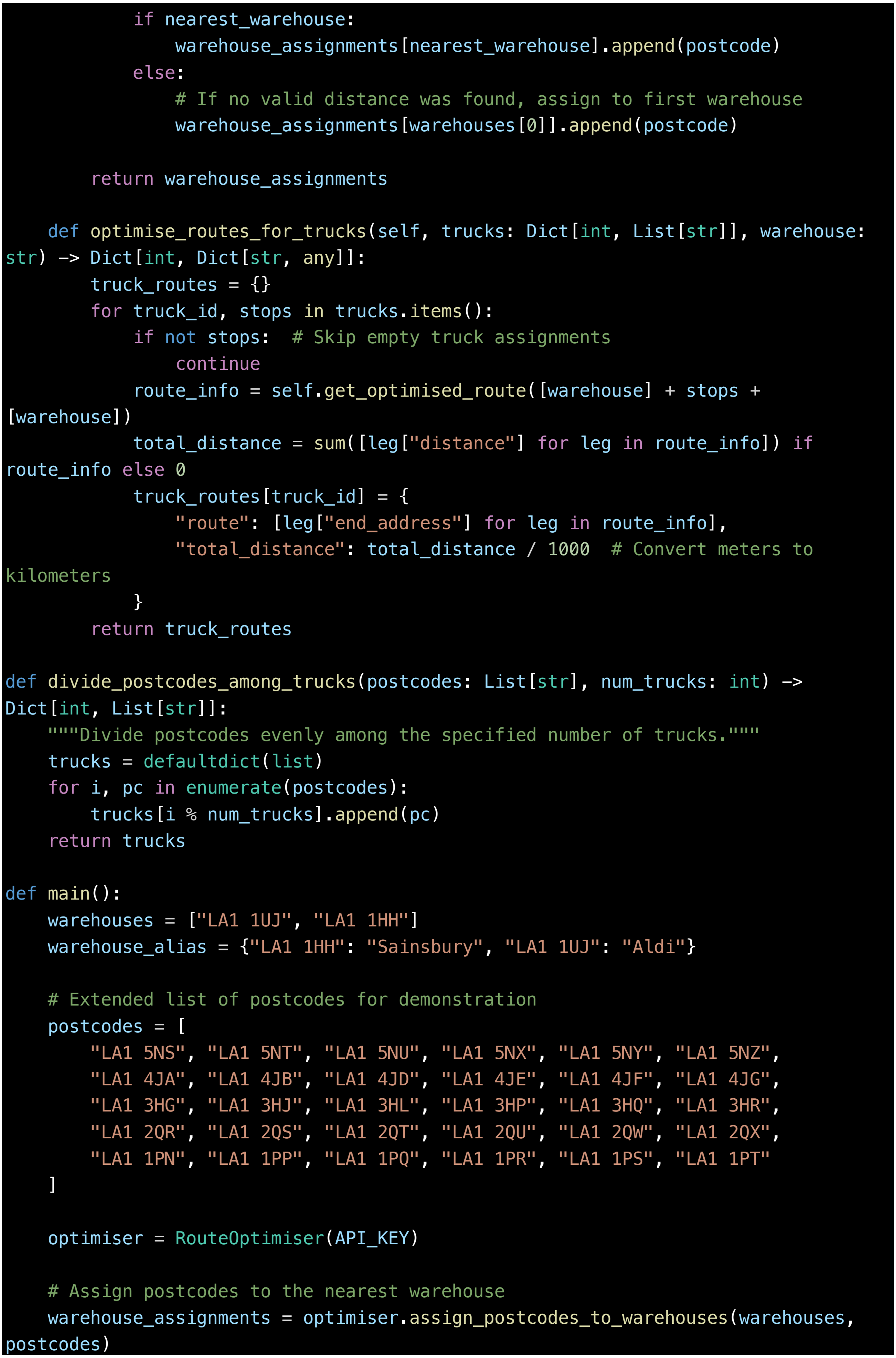
Appendix 1.9: Showcases the food distribution for warehouses.

Appendix 1.10: Python code for VRP problem, this problem calculated for 30 postcodes.

```
import os
import requests
import statistics
import logging
from concurrent.futures import ThreadPoolExecutor, as_completed
from collections import defaultdict
from typing import List, Dict, Optional, Tuple

# Configure logging
logging.basicConfig(level=logging.INFO)
logger = logging.getLogger(__name__)

# Load API key from environment variable
API_KEY = ("AIzaSyD6pekLFITfzOyBYY35-vf1QcaTfZ0GY24")


class RouteOptimiser:
    def __init__(self, api_key: str):
        self.api_key = api_key

    def get_distance(self, origin: str, destination: str) -> Optional[int]:
        url = (
            f"https://maps.googleapis.com/maps/api/distancematrix/json"

f"?origins={origin}&destinations={destination}&mode=driving&key={self.api_key}"
        )
        try:
            response = requests.get(url)
            response.raise_for_status()
            data = response.json()
            if data["status"] == "OK" and data["rows"]:
                element = data["rows"][0]["elements"][0]
                if element["status"] == "OK" and "distance" in element:
                    return element["distance"]["value"]
        except requests.exceptions.RequestException as e:
            logger.error(f"Error retrieving distance from {origin} to
{destination}: {e}")
        return None

    def get_optimised_route(self, waypoints: List[str]) -> List[Dict[str, str]]:
        if not waypoints:
            return []
        origin = waypoints[0]
        destination = waypoints[-1]
        stops = waypoints[1:-1]
        waypoints_str = "|".join(stops)
        url = (
            f"https://maps.googleapis.com/maps/api/directions/json?"
```



```
f"origin={origin}&destination={destination}&waypoints=optimise:true|{waypoints_str}
&key={self.api_key}"
        )
        try:
            response = requests.get(url)
            response.raise_for_status()
            data = response.json()
            if data["status"] == "OK":
                route_info = []
                for leg in data["routes"][0]["legs"]:
                    route_info.append({
                        "end_address": leg["end_address"],
                        "distance": leg["distance"]["value"]
                    })
                return route_info
        except requests.exceptions.RequestException as e:
            logger.error(f"Error retrieving optimised route: {e}")
        return []

    def calculate_warehouse_distances(self, warehouses: List[str], postcodes:
List[str]) -> Dict[str, Dict[str, int]]:
        """Calculate distances from each warehouse to each postcode."""
        warehouse_distances = {warehouse: {} for warehouse in warehouses}

        for warehouse in warehouses:
            for postcode in postcodes:
                distance = self.get_distance(warehouse, postcode)
                if distance is not None:
                    warehouse_distances[warehouse][postcode] = distance

        return warehouse_distances

    def assign_postcodes_to_warehouses(self, warehouses: List[str], postcodes:
List[str]) -> Dict[str, List[str]]:
        """Assign each postcode to the nearest warehouse."""
        warehouse_distances = self.calculate_warehouse_distances(warehouses,
postcodes)
        warehouse_assignments = {warehouse: [] for warehouse in warehouses}

        for postcode in postcodes:
            min_distance = float('inf')
            nearest_warehouse = None

            for warehouse in warehouses:
                if postcode in warehouse_distances[warehouse]:
                    distance = warehouse_distances[warehouse][postcode]
                    if distance < min_distance:
                        min_distance = distance
                        nearest_warehouse = warehouse
```

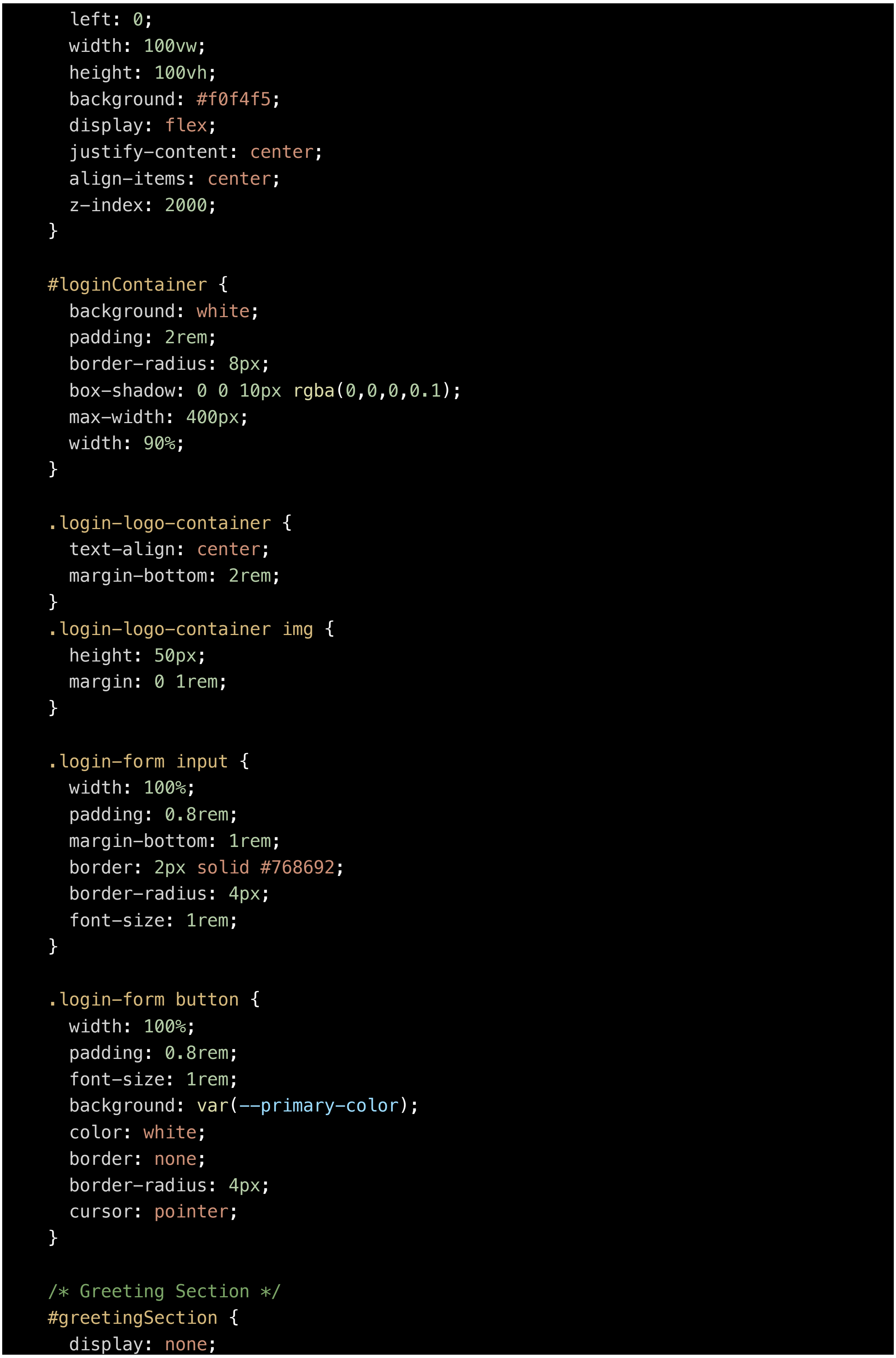


```
            if nearest_warehouse:
                warehouse_assignments[nearest_warehouse].append(postcode)
            else:
                # If no valid distance was found, assign to first warehouse
                warehouse_assignments[warehouses[0]].append(postcode)

        return warehouse_assignments

    def optimise_routes_for_trucks(self, trucks: Dict[int, List[str]], warehouse: 
str) -> Dict[int, Dict[str, any]]:
        truck_routes = {}
        for truck_id, stops in trucks.items():
            if not stops:  # Skip empty truck assignments
                continue
            route_info = self.get_optimised_route([warehouse] + stops + 
[warehouse])
            total_distance = sum([leg["distance"] for leg in route_info]) if 
route_info else 0
            truck_routes[truck_id] = {
                "route": [leg["end_address"] for leg in route_info],
                "total_distance": total_distance / 1000  # Convert meters to 
kilometers
            }
        return truck_routes

def divide_postcodes_among_trucks(postcodes: List[str], num_trucks: int) -> 
Dict[int, List[str]]:
    """Divide postcodes evenly among the specified number of trucks."""
    trucks = defaultdict(list)
    for i, pc in enumerate(postcodes):
        trucks[i % num_trucks].append(pc)
    return trucks

def main():
    warehouses = ["LA1 1UJ", "LA1 1HH"]
    warehouse_alias = {"LA1 1HH": "Sainsbury", "LA1 1UJ": "Aldi"}

    # Extended list of postcodes for demonstration
    postcodes = [
        "LA1 5NS", "LA1 5NT", "LA1 5NU", "LA1 5NX", "LA1 5NY", "LA1 5NZ",
        "LA1 4JA", "LA1 4JB", "LA1 4JD", "LA1 4JE", "LA1 4JF", "LA1 4JG",
        "LA1 3HG", "LA1 3HJ", "LA1 3HL", "LA1 3HP", "LA1 3HQ", "LA1 3HR",
        "LA1 2QR", "LA1 2QS", "LA1 2QT", "LA1 2QU", "LA1 2QW", "LA1 2QX",
        "LA1 1PN", "LA1 1PP", "LA1 1PQ", "LA1 1PR", "LA1 1PS", "LA1 1PT"
    ]

    optimiser = RouteOptimiser(API_KEY)

    # Assign postcodes to the nearest warehouse
    warehouse_assignments = optimiser.assign_postcodes_to_warehouses(warehouses, 
postcodes)
```

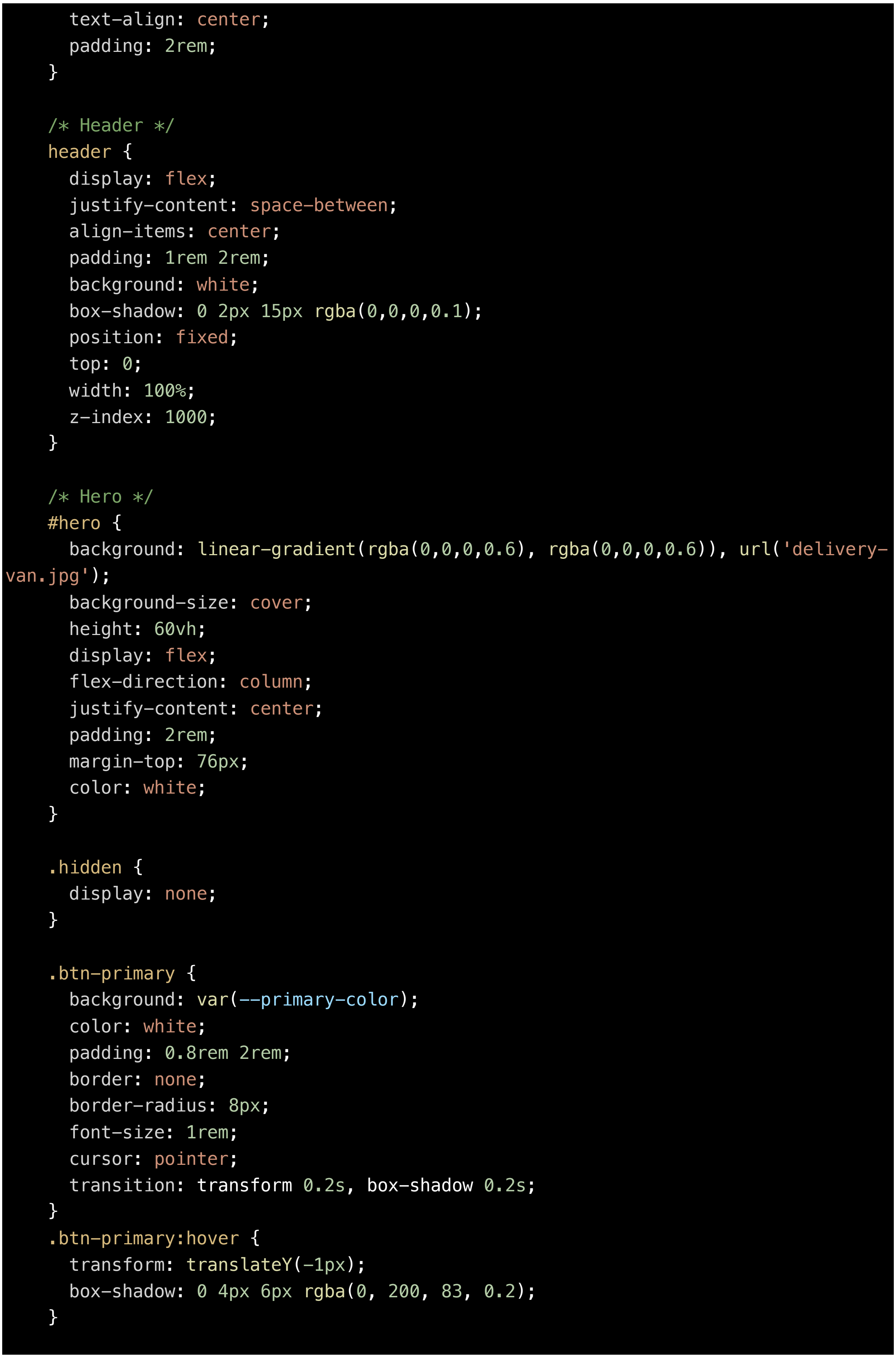
```
    # Total number of trucks is 30
    total_trucks = 30

    # Calculate how many trucks to assign to each warehouse based on the proportion
of postcodes
    total_postcodes = len(postcodes)
    trucks_per_warehouse = {}

    for warehouse in warehouses:
        assigned_postcodes = len(warehouse_assignments[warehouse])
        proportion = assigned_postcodes / total_postcodes
        trucks_per_warehouse[warehouse] = max(1, round(proportion * total_trucks))

    # Adjust if total is not exactly 30
    trucks_diff = total_trucks - sum(trucks_per_warehouse.values())
    if trucks_diff != 0:
        # Add/remove trucks from the warehouse with more postcodes
        if len(warehouse_assignments[warehouses[0]]) >
len(warehouse_assignments[warehouses[1]]):
            trucks_per_warehouse[warehouses[0]] += trucks_diff
        else:
            trucks_per_warehouse[warehouses[1]] += trucks_diff

    # Now assign trucks for each warehouse
    all_truck_routes = {}
    grand_total_distance = 0

    # Track global truck IDs
    global_truck_id = 0

    for warehouse in warehouses:
        warehouse_label = warehouse_alias.get(warehouse, warehouse)
        assigned_postcodes = warehouse_assignments[warehouse]
        num_trucks = trucks_per_warehouse[warehouse]

        # Skip if no postcodes assigned to this warehouse
        if not assigned_postcodes:
            continue

        # Divide postcodes among trucks for this warehouse
        trucks = divide_postcodes_among_trucks(assigned_postcodes, num_trucks)

        # Optimise routes for each truck
        truck_routes = optimiser.optimise_routes_for_trucks(trucks, warehouse)

        print("========================================================")
        print(f"Warehouse: {warehouse_label} (Assigned {num_trucks} trucks)")
        print("========================================================")

        warehouse_total_distance = 0
        for local_truck_id, data in truck_routes.items():
            actual_truck_id = global_truck_id + 1
            global_truck_id += 1

            distance = data['total_distance']
            warehouse_total_distance += distance
            grand_total_distance += distance

            print(f"Truck {actual_truck_id} Route (Total Distance: {distance:.2f}
km):")
            for stop in data['route']:
                print(f"  {stop}")
            print("------------------------------------------------")

        print(f"Total distance for {warehouse_label}:
{warehouse_total_distance:.2f} km")
        print("========================================================")

    print(f"Grand total distance across all trucks: {grand_total_distance:.2f} km")

if __name__ == "__main__":
    main()
```

Appendix 1.11: Website HTML Code (just frontend)

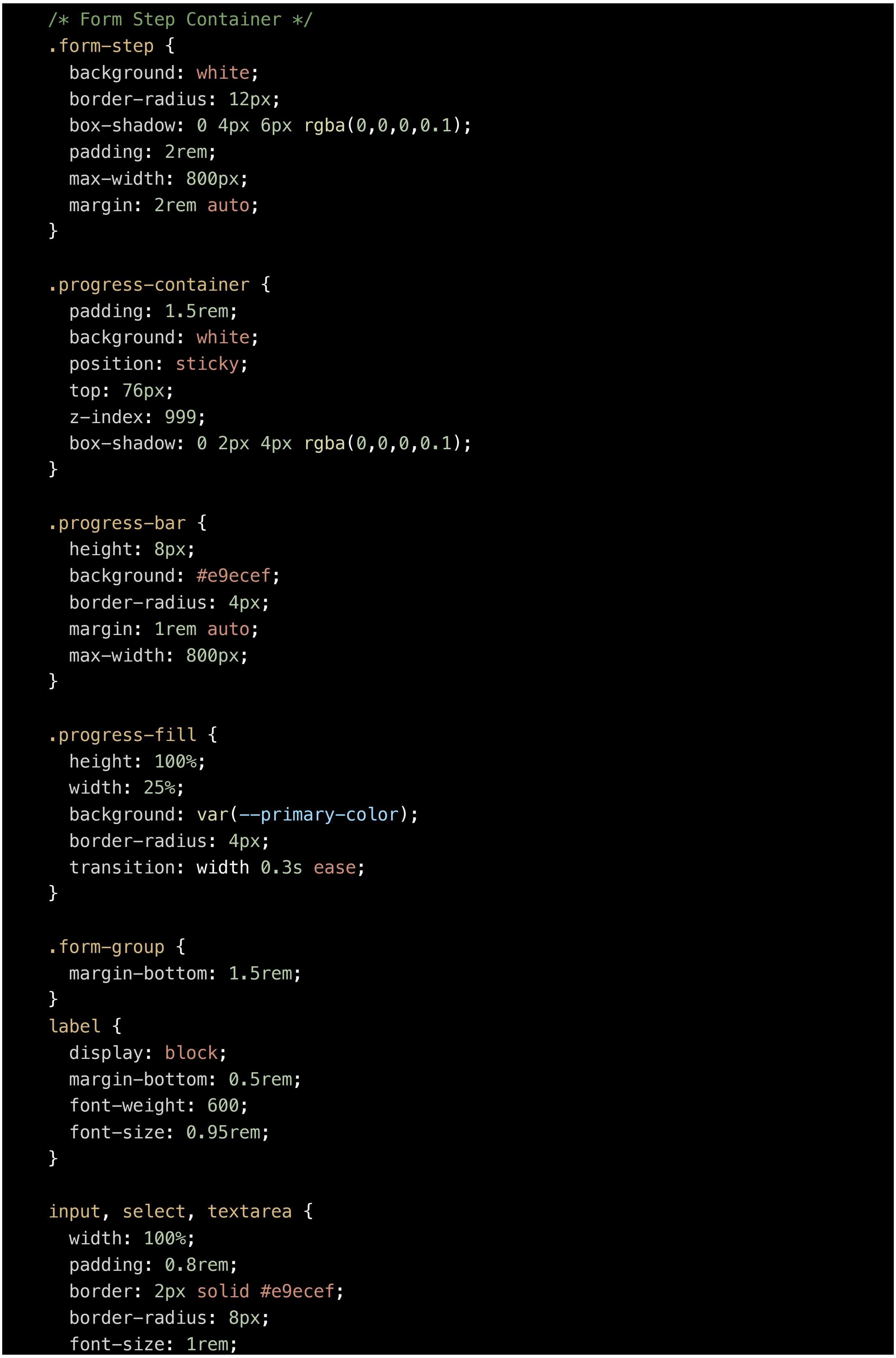
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
  <title>Lancaster Food Delivery</title>
  <style>
    :root {
      --primary-color: #00C853;
      --secondary-color: #f5f5f5;
      --text-color: #333;
      --error-color: #e74c3c;
    }

    body {
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
      margin: 0;
      background-color: #f8f9fa;
      color: var(--text-color);
    }

    /* Login Overlay */
    #loginOverlay {
      position: fixed;
      top: 0;
```



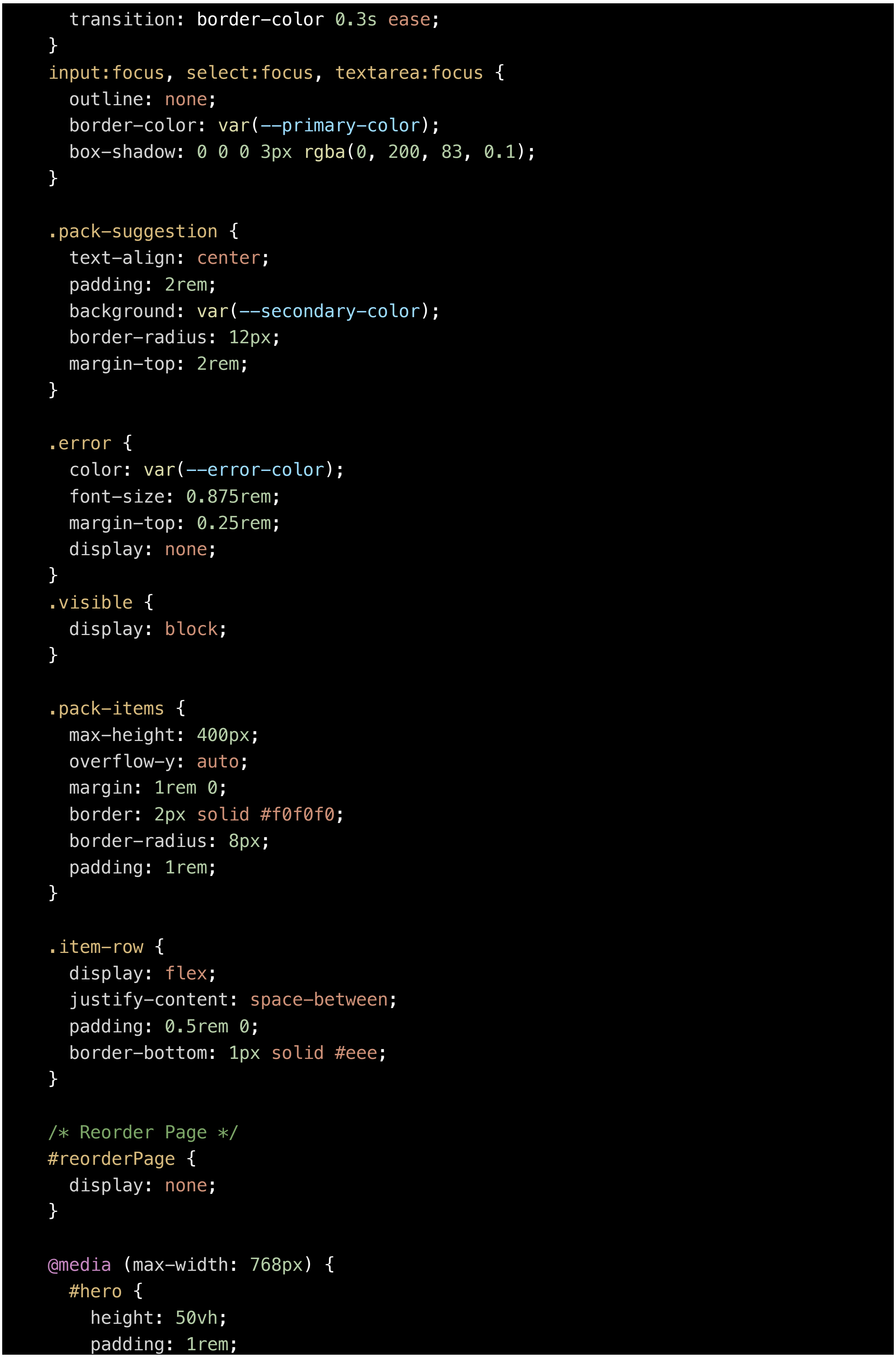
```
    left: 0;
    width: 100vw;
    height: 100vh;
    background: #f0f4f5;
    display: flex;
    justify-content: center;
    align-items: center;
    z-index: 2000;
  }

  #loginContainer {
    background: white;
    padding: 2rem;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0,0,0,0.1);
    max-width: 400px;
    width: 90%;
  }

  .login-logo-container {
    text-align: center;
    margin-bottom: 2rem;
  }
  .login-logo-container img {
    height: 50px;
    margin: 0 1rem;
  }

  .login-form input {
    width: 100%;
    padding: 0.8rem;
    margin-bottom: 1rem;
    border: 2px solid #768692;
    border-radius: 4px;
    font-size: 1rem;
  }

  .login-form button {
    width: 100%;
    padding: 0.8rem;
    font-size: 1rem;
    background: var(--primary-color);
    color: white;
    border: none;
    border-radius: 4px;
    cursor: pointer;
  }

  /* Greeting Section */
  #greetingSection {
    display: none;
```



```
      text-align: center;
      padding: 2rem;
    }

    /* Header */
    header {
      display: flex;
      justify-content: space-between;
      align-items: center;
      padding: 1rem 2rem;
      background: white;
      box-shadow: 0 2px 15px rgba(0,0,0,0.1);
      position: fixed;
      top: 0;
      width: 100%;
      z-index: 1000;
    }

    /* Hero */
    #hero {
      background: linear-gradient(rgba(0,0,0,0.6), rgba(0,0,0,0.6)), url('delivery-van.jpg');
      background-size: cover;
      height: 60vh;
      display: flex;
      flex-direction: column;
      justify-content: center;
      padding: 2rem;
      margin-top: 76px;
      color: white;
    }

    .hidden {
      display: none;
    }

    .btn-primary {
      background: var(--primary-color);
      color: white;
      padding: 0.8rem 2rem;
      border: none;
      border-radius: 8px;
      font-size: 1rem;
      cursor: pointer;
      transition: transform 0.2s, box-shadow 0.2s;
    }
    .btn-primary:hover {
      transform: translateY(-1px);
      box-shadow: 0 4px 6px rgba(0, 200, 83, 0.2);
    }
```



```
    /* Form Step Container */
    .form-step {
      background: white;
      border-radius: 12px;
      box-shadow: 0 4px 6px rgba(0,0,0,0.1);
      padding: 2rem;
      max-width: 800px;
      margin: 2rem auto;
    }

    .progress-container {
      padding: 1.5rem;
      background: white;
      position: sticky;
      top: 76px;
      z-index: 999;
      box-shadow: 0 2px 4px rgba(0,0,0,0.1);
    }

    .progress-bar {
      height: 8px;
      background: #e9ecef;
      border-radius: 4px;
      margin: 1rem auto;
      max-width: 800px;
    }

    .progress-fill {
      height: 100%;
      width: 25%;
      background: var(--primary-color);
      border-radius: 4px;
      transition: width 0.3s ease;
    }

    .form-group {
      margin-bottom: 1.5rem;
    }
    label {
      display: block;
      margin-bottom: 0.5rem;
      font-weight: 600;
      font-size: 0.95rem;
    }

    input, select, textarea {
      width: 100%;
      padding: 0.8rem;
      border: 2px solid #e9ecef;
      border-radius: 8px;
      font-size: 1rem;
```

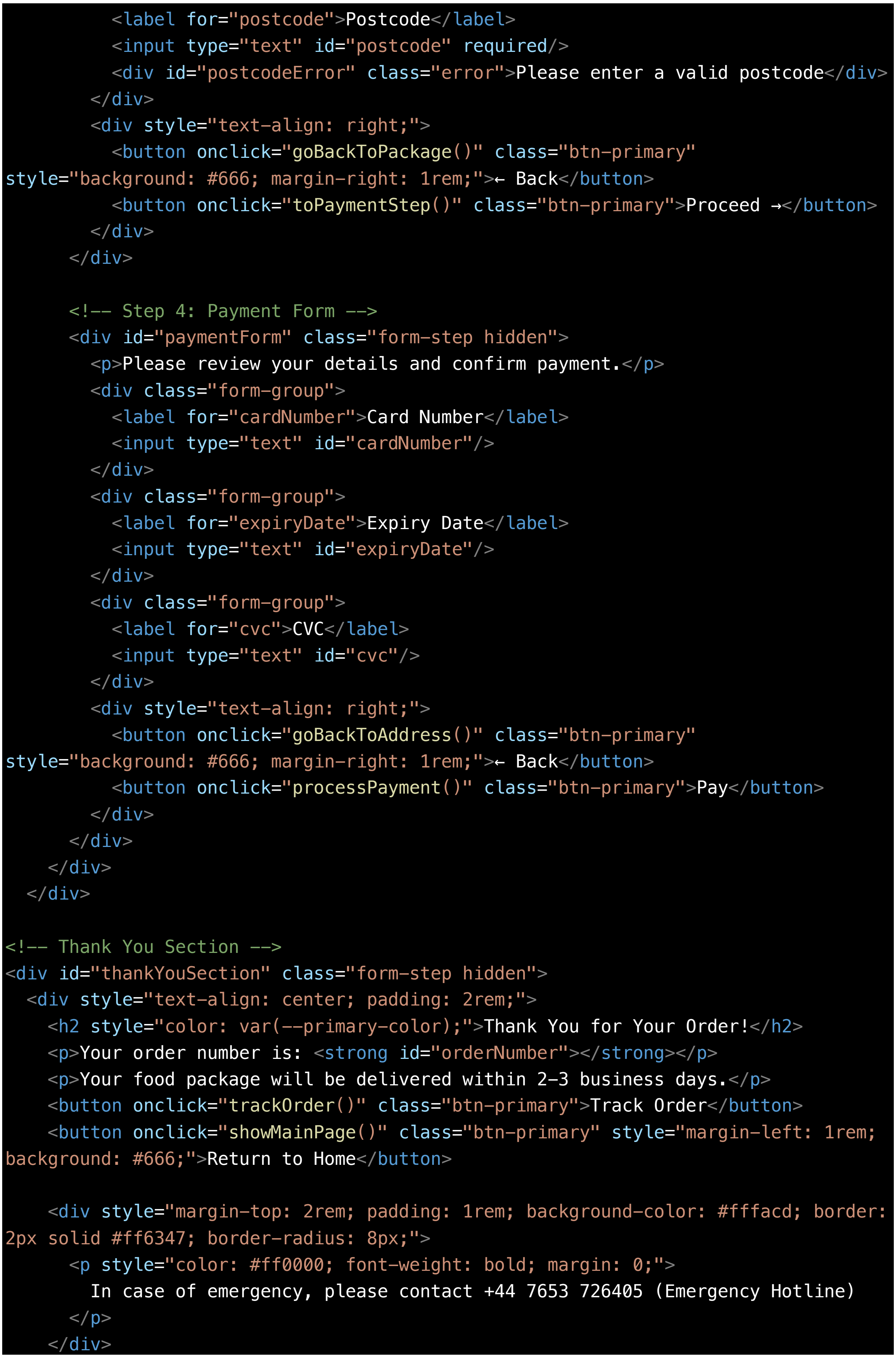


```
  transition: border-color 0.3s ease;
}
input:focus, select:focus, textarea:focus {
  outline: none;
  border-color: var(--primary-color);
  box-shadow: 0 0 0 3px rgba(0, 200, 83, 0.1);
}

.pack-suggestion {
  text-align: center;
  padding: 2rem;
  background: var(--secondary-color);
  border-radius: 12px;
  margin-top: 2rem;
}

.error {
  color: var(--error-color);
  font-size: 0.875rem;
  margin-top: 0.25rem;
  display: none;
}
.visible {
  display: block;
}

.pack-items {
  max-height: 400px;
  overflow-y: auto;
  margin: 1rem 0;
  border: 2px solid #f0f0f0;
  border-radius: 8px;
  padding: 1rem;
}

.item-row {
  display: flex;
  justify-content: space-between;
  padding: 0.5rem 0;
  border-bottom: 1px solid #eee;
}

/* Reorder Page */
#reorderPage {
  display: none;
}

@media (max-width: 768px) {
  #hero {
    height: 50vh;
    padding: 1rem;
```

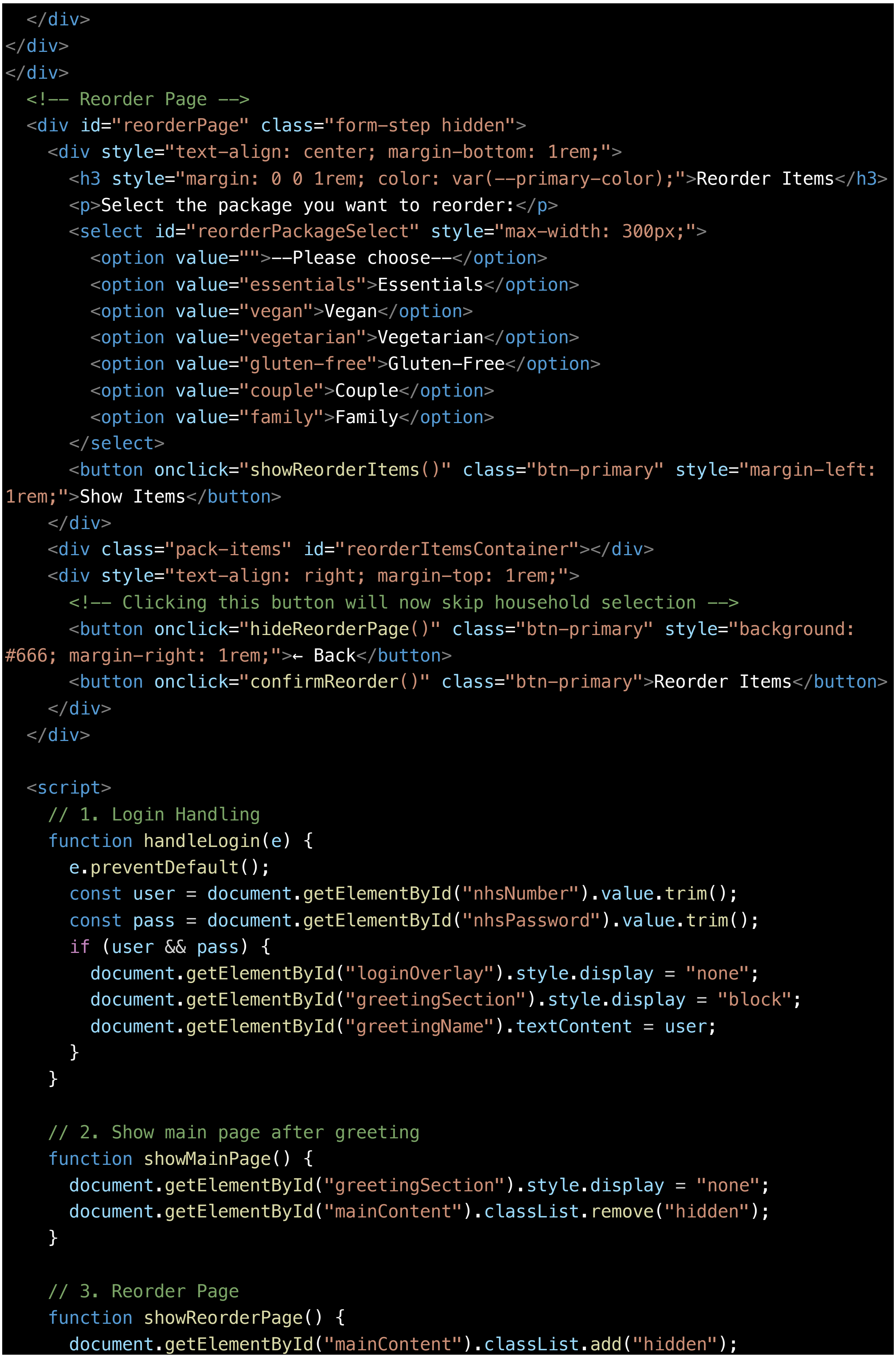
```
      }
      #hero h1 {
        font-size: 2rem;
      }
      .form-step {
        padding: 1.5rem;
        margin: 1rem;
      }
    }
  </style>
</head>
<body>
  <!-- Login Overlay -->
  <div id="loginOverlay">
    <div id="loginContainer">
      <div class="login-logo-container">
        <img src="NHS.png" alt="NHS Logo">
        <img src="lancaster-logo.png" alt="Lancaster City Council">
      </div>
      <form class="login-form" onsubmit="handleLogin(event)">
        <input type="text" id="nhsNumber" placeholder="NHS Login ID" required/>
        <input type="password" id="nhsPassword" placeholder="Password" required/>
        <button type="submit">Log in with NHS</button>
      </form>
    </div>
  </div>

  <!-- Greeting Section -->
  <div id="greetingSection">
    <h2>Welcome, <span id="greetingName"></span>!</h2>
    <p>You have successfully logged in. Click to proceed to the homepage.</p>
    <button class="btn-primary" onclick="showMainPage()">Go to Homepage</button>
  </div>

  <!-- Main Content (hidden by default) -->
  <div id="mainContent" class="hidden">
    <header>
      <img src="lancaster-logo.png" alt="Lancaster City Council" style="height: 
40px;">
      <div style="display: flex; gap: 20px;">
        <!-- Profile icon triggers reorder page -->
        <img src="profile.png" alt="Profile" style="height: 24px; cursor:pointer;" 
onclick="showReorderPage()">
      </div>
    </header>

    <!-- Hero -->
    <section id="hero">
      <h1 style="font-size: 2.5rem; margin-bottom: 1rem;">Delivering 
Essentials</h1>
```

```
   <p style="font-size: 1.25rem; margin-bottom: 2rem;">Safely to Your 
Doorstep</p>
   <button onclick="showOrderContent()" class="btn-primary">Place an 
Order</button>
   <!-- Users can click here to reorder as well -->
   <button onclick="showReorderPage()" class="btn-primary" style="margin-top: 
1rem;">Reorder Items</button>
  </section>

  <!-- Main Container (all steps) -->
  <div id="orderContent" class="hidden">
   <!-- Progress Container -->
   <div class="progress-container">
    <div class="progress-bar">
     <div class="progress-fill"></div>
    </div>
    <p id="progressText" style="text-align: center; margin: 0.5rem 0;">
     Step 1 of 4 - Household Information
    </p>
   </div>

   <!-- Step 1: Household Info -->
   <div id="householdInfoStep" class="form-step">
    <form onsubmit="return suggestPack(event)">
     <div class="form-group">
      <label for="householdSize">Number of people in household</label>
      <input
       type="number"
       id="householdSize"
       min="1"
       required
       oninput="createAgeFields(this.value)"
      />
      <div id="householdError" class="error">Please enter a valid 
number</div>
     </div>

     <div id="ageFieldsContainer" class="form-group"></div>

     <div class="form-group">
      <label>Dietary Requirements</label>
      <select id="dietaryRequirements" required>
       <option value="N/A">No special requirements</option>
       <option value="vegan">Vegan</option>
       <option value="vegetarian">Vegetarian</option>
       <option value="gluten-free">Gluten-free</option>
      </select>
     </div>

     <div class="form-group">
      <label for="concerns">Additional Notes</label>
```

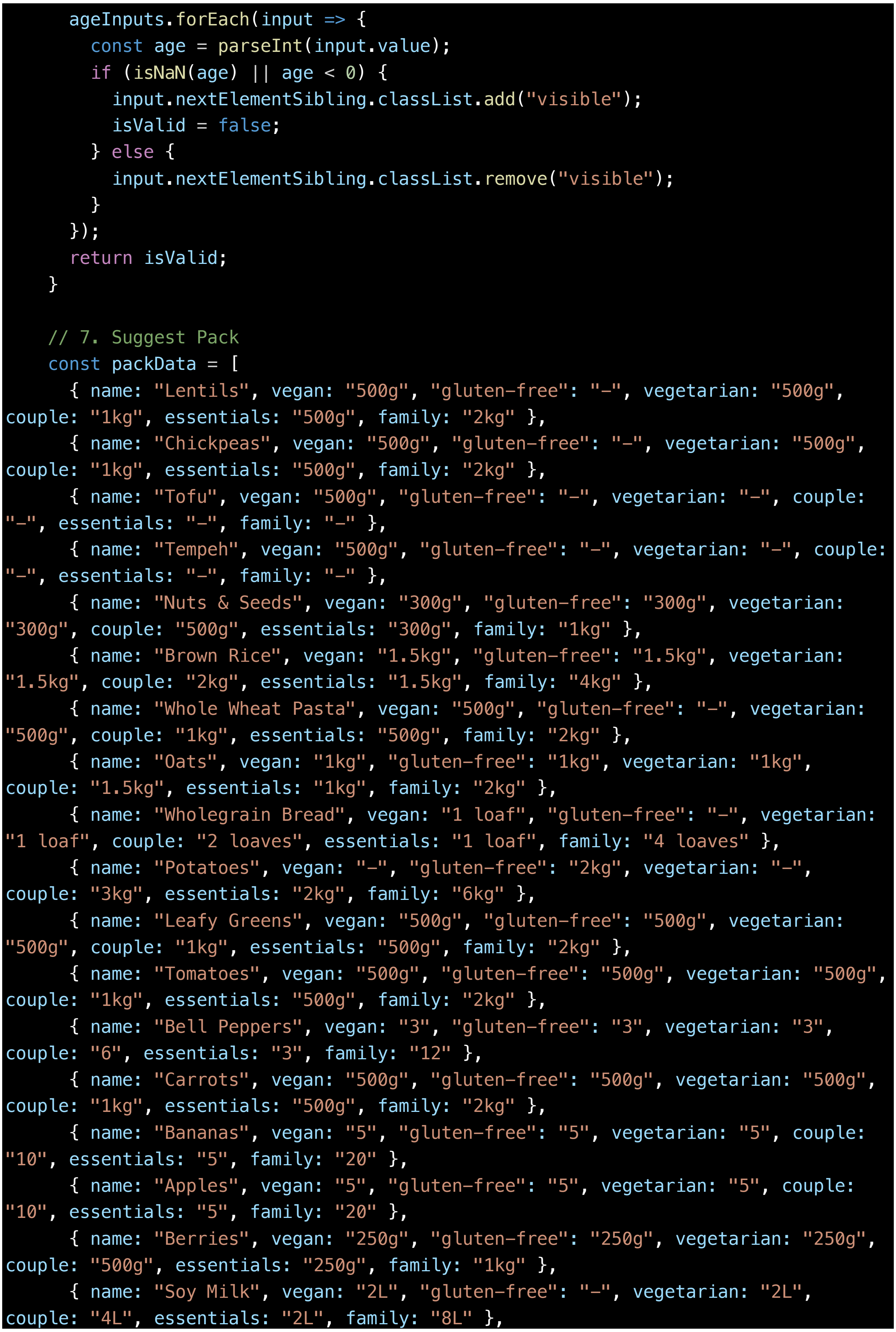
```
            <textarea id="concerns" rows="4" placeholder="Any special requests or
concerns"></textarea>
          </div>

          <div style="text-align: right;">
            <button type="submit" class="btn-primary">Continue →</button>
          </div>
        </form>
      </div>

      <!-- Step 2: Package Selection -->
      <div id="packSuggestion" class="form-step hidden">
        <div class="pack-suggestion">
          <h3 id="packTitle" style="margin: 0 0 1rem; color: var(--primary-
color);"></h3>
          <div class="pack-items" id="packItems"></div>
          <small style="display: block; color: #999; margin-top: 1rem;">* These
quantities should last about a week</small>
          <div style="margin-top: 2rem;">
            <button onclick="confirmSelection()" class="btn-primary">Confirm
Selection</button>
            <button onclick="goBackToHousehold()" class="btn-primary"
style="background: #666; margin-left: 1rem;">← Back</button>
          </div>
        </div>
      </div>

      <!-- Step 3: Address Form -->
      <div id="addressForm" class="form-step hidden">
        <div class="form-group">
          <label for="houseNumber">House or Flat Number</label>
          <input type="text" id="houseNumber" required/>
          <div id="houseNumberError" class="error">Please enter a valid house/flat
number</div>
        </div>
        <div class="form-group">
          <label for="streetAddress1">Street Address 1</label>
          <input type="text" id="streetAddress1" required/>
          <div id="streetAddress1Error" class="error">Please enter a valid street
address</div>
        </div>
        <div class="form-group">
          <label for="streetAddress2">Street Address 2 (optional)</label>
          <input type="text" id="streetAddress2"/>
        </div>
        <div class="form-group">
          <label for="city">City/Town</label>
          <input type="text" id="city" required/>
          <div id="cityError" class="error">Please enter a valid city/town</div>
        </div>
        <div class="form-group">
```



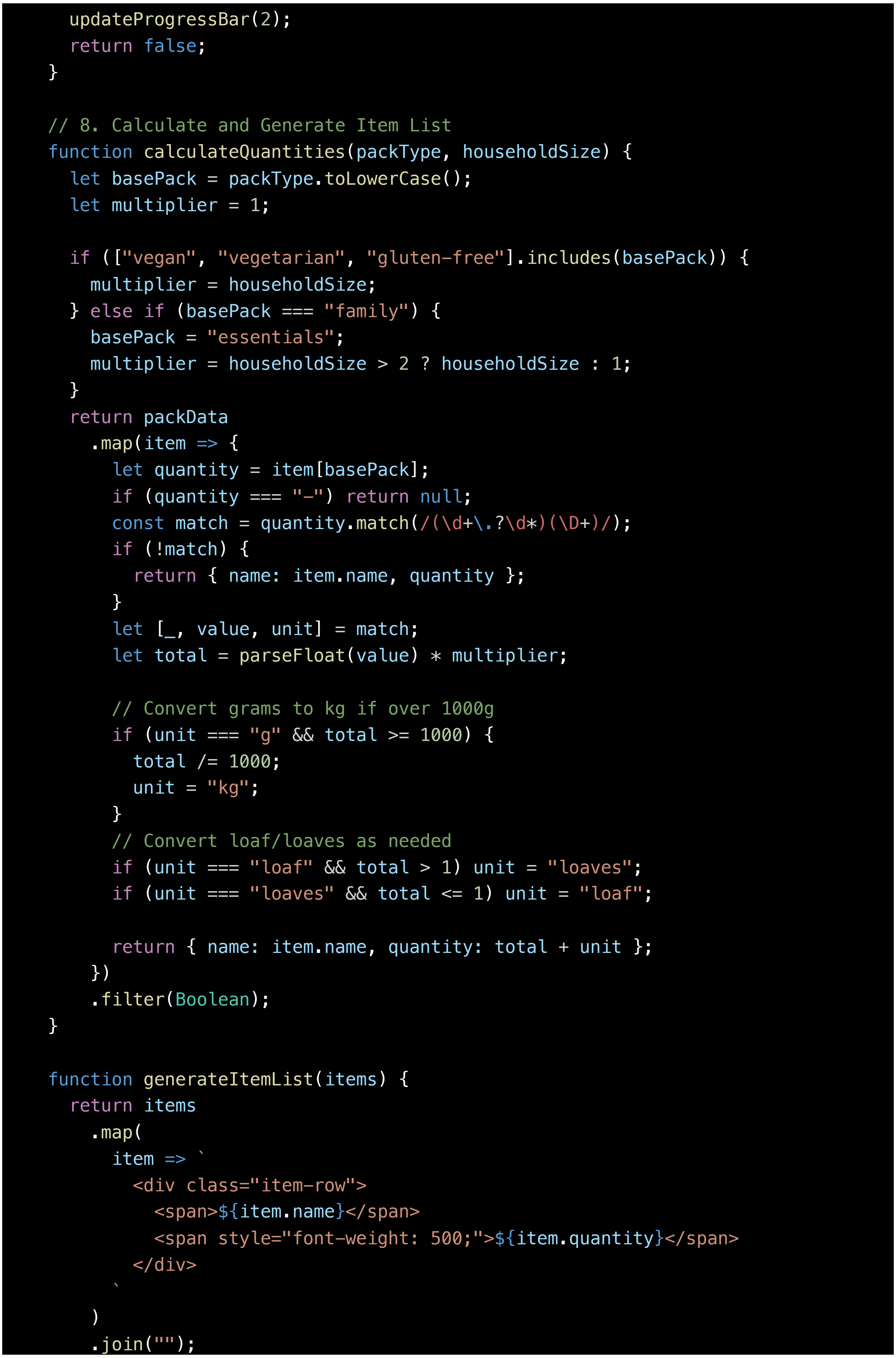
```
          <label for="postcode">Postcode</label>
          <input type="text" id="postcode" required/>
          <div id="postcodeError" class="error">Please enter a valid postcode</div>
        </div>
        <div style="text-align: right;">
          <button onclick="goBackToPackage()" class="btn-primary"
style="background: #666; margin-right: 1rem;">← Back</button>
          <button onclick="toPaymentStep()" class="btn-primary">Proceed →</button>
        </div>
      </div>

      <!-- Step 4: Payment Form -->
      <div id="paymentForm" class="form-step hidden">
        <p>Please review your details and confirm payment.</p>
        <div class="form-group">
          <label for="cardNumber">Card Number</label>
          <input type="text" id="cardNumber"/>
        </div>
        <div class="form-group">
          <label for="expiryDate">Expiry Date</label>
          <input type="text" id="expiryDate"/>
        </div>
        <div class="form-group">
          <label for="cvc">CVC</label>
          <input type="text" id="cvc"/>
        </div>
        <div style="text-align: right;">
          <button onclick="goBackToAddress()" class="btn-primary"
style="background: #666; margin-right: 1rem;">← Back</button>
          <button onclick="processPayment()" class="btn-primary">Pay</button>
        </div>
      </div>
    </div>
  </div>

<!-- Thank You Section -->
<div id="thankYouSection" class="form-step hidden">
  <div style="text-align: center; padding: 2rem;">
    <h2 style="color: var(--primary-color);">Thank You for Your Order!</h2>
    <p>Your order number is: <strong id="orderNumber"></strong></p>
    <p>Your food package will be delivered within 2-3 business days.</p>
    <button onclick="trackOrder()" class="btn-primary">Track Order</button>
    <button onclick="showMainPage()" class="btn-primary" style="margin-left: 1rem;
background: #666;">Return to Home</button>

    <div style="margin-top: 2rem; padding: 1rem; background-color: #fffacd; border:
2px solid #ff6347; border-radius: 8px;">
      <p style="color: #ff0000; font-weight: bold; margin: 0;">
        In case of emergency, please contact +44 7653 726405 (Emergency Hotline)
      </p>
    </div>
```



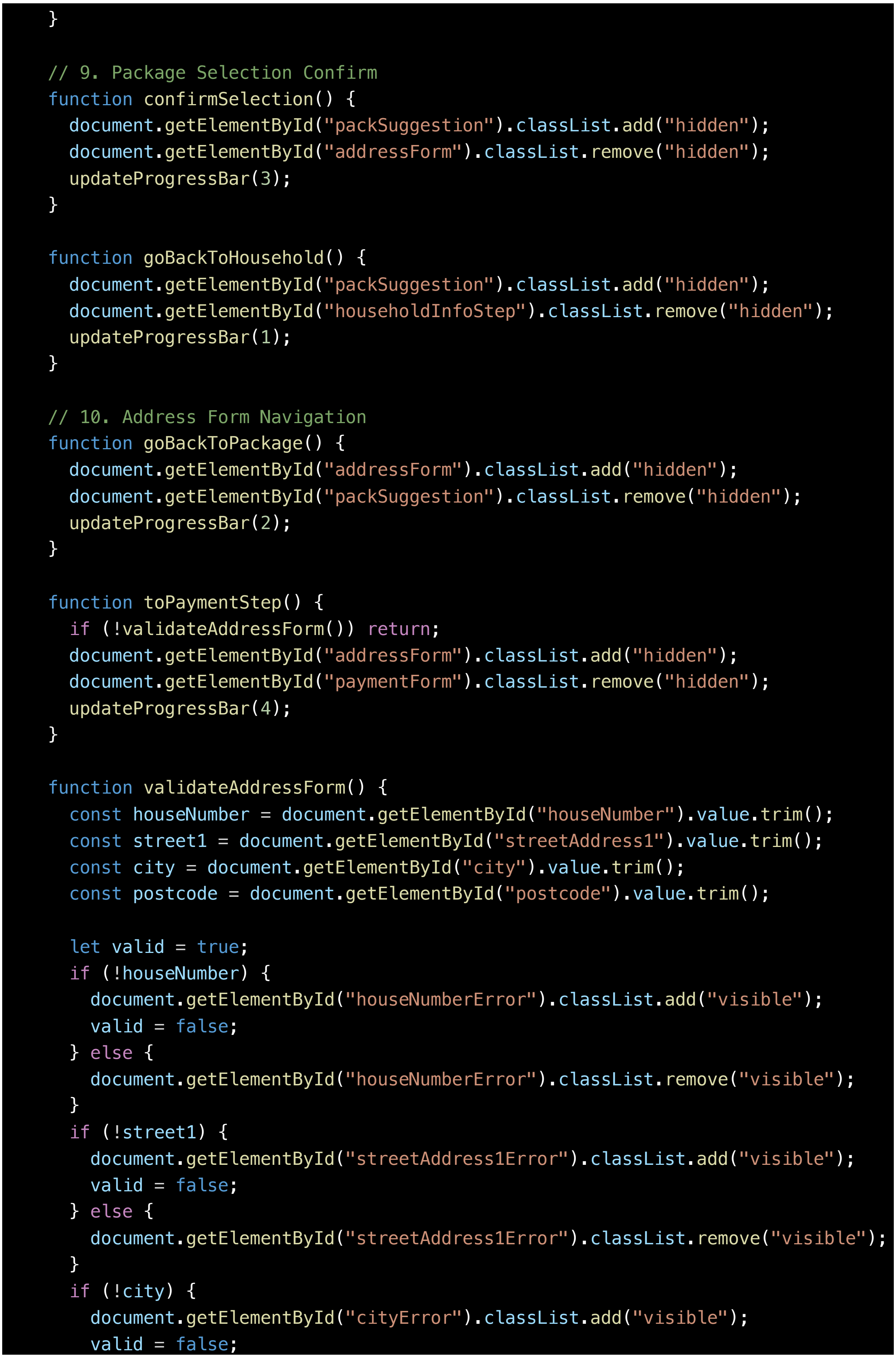
```
  </div>
</div>
</div>
  <!-- Reorder Page -->
  <div id="reorderPage" class="form-step hidden">
    <div style="text-align: center; margin-bottom: 1rem;">
      <h3 style="margin: 0 0 1rem; color: var(--primary-color);">Reorder Items</h3>
      <p>Select the package you want to reorder:</p>
      <select id="reorderPackageSelect" style="max-width: 300px;">
        <option value="">--Please choose--</option>
        <option value="essentials">Essentials</option>
        <option value="vegan">Vegan</option>
        <option value="vegetarian">Vegetarian</option>
        <option value="gluten-free">Gluten-Free</option>
        <option value="couple">Couple</option>
        <option value="family">Family</option>
      </select>
      <button onclick="showReorderItems()" class="btn-primary" style="margin-left:
1rem;">Show Items</button>
    </div>
    <div class="pack-items" id="reorderItemsContainer"></div>
    <div style="text-align: right; margin-top: 1rem;">
      <!-- Clicking this button will now skip household selection -->
      <button onclick="hideReorderPage()" class="btn-primary" style="background:
#666; margin-right: 1rem;">← Back</button>
      <button onclick="confirmReorder()" class="btn-primary">Reorder Items</button>
    </div>
  </div>

  <script>
    // 1. Login Handling
    function handleLogin(e) {
      e.preventDefault();
      const user = document.getElementById("nhsNumber").value.trim();
      const pass = document.getElementById("nhsPassword").value.trim();
      if (user && pass) {
        document.getElementById("loginOverlay").style.display = "none";
        document.getElementById("greetingSection").style.display = "block";
        document.getElementById("greetingName").textContent = user;
      }
    }

    // 2. Show main page after greeting
    function showMainPage() {
      document.getElementById("greetingSection").style.display = "none";
      document.getElementById("mainContent").classList.remove("hidden");
    }

    // 3. Reorder Page
    function showReorderPage() {
      document.getElementById("mainContent").classList.add("hidden");
      document.getElementById("reorderPage").classList.remove("hidden");
      document.getElementById("reorderPackageSelect").value = "";
      document.getElementById("reorderItemsContainer").innerHTML = "";
    }

    function hideReorderPage() {
      document.getElementById("reorderPage").classList.add("hidden");
      document.getElementById("mainContent").classList.remove("hidden");
    }

    function showReorderItems() {
      const packageSelect = document.getElementById("reorderPackageSelect");
      const selectedPackage = packageSelect.value;
      const container = document.getElementById("reorderItemsContainer");
      container.innerHTML = "";

      if (!selectedPackage) {
        container.innerHTML = '<p style="color: var(--error-color);">Please select
a package first.</p>';
        return;
      }

      const itemList = calculateQuantities(selectedPackage, 1);
      const checkboxesHtml = itemList
        .map(
          item => `
            <div class="item-row">
              <label style="display: flex; align-items: center;">
                <input type="checkbox" name="reorderItems" value="${item.name}">
                <span style="margin-left: 8px;">${item.name}
(${item.quantity})</span>
              </label>
            </div>
          `
        )
        .join("");
      container.innerHTML = checkboxesHtml;
    }

    // 4. Confirm Reorder: Skip household info, go straight to address form
    function confirmReorder() {
      const checkedBoxes =
document.querySelectorAll('input[name="reorderItems"]:checked');
      if (!checkedBoxes.length) {
        alert("No items selected for reorder.");
        return;
      }
      const selectedItems = [];
      checkedBoxes.forEach(box => selectedItems.push(box.value));
      alert("You have chosen to reorder: " + selectedItems.join(", "));
      // Hide the reorder page
      document.getElementById("reorderPage").classList.add("hidden");

      // Reveal the order content container, and skip to Address Form (step 3)
      document.getElementById("orderContent").classList.remove("hidden");

      // Hide steps 1 and 2 so they are not visible in a reorder flow
      document.getElementById("householdInfoStep").classList.add("hidden");
      document.getElementById("packSuggestion").classList.add("hidden");

      // Show Address Form (step 3) for the delivery portal
      document.getElementById("addressForm").classList.remove("hidden");
      updateProgressBar(3);
    }

    // 5. Show or hide main order content
    function showOrderContent() {
      document.getElementById("orderContent").classList.remove("hidden");
      document.getElementById("orderContent").scrollIntoView({ behavior: "smooth"
});
      updateProgressBar(1);
    }

    // 6. Household Step
    function createAgeFields(number) {
      const container = document.getElementById("ageFieldsContainer");
      container.innerHTML = "";
      if (number < 1) return;
      for (let i = 1; i <= number; i++) {
        const div = document.createElement("div");
        div.className = "form-group";
        div.innerHTML = `
          <label for="person${i}Age">Person ${i} Age</label>
          <input type="number" id="person${i}Age" min="0" required>
          <div class="error" id="ageError${i}">Please enter a valid age</div>
        `;
        container.appendChild(div);
      }
    }

    function validateForm() {
      const householdSize =
parseInt(document.getElementById("householdSize").value);
      if (isNaN(householdSize) || householdSize < 1) {
        document.getElementById("householdError").classList.add("visible");
        return false;
      }
      document.getElementById("householdError").classList.remove("visible");

      const ageInputs = document.querySelectorAll("#ageFieldsContainer input");
      let isValid = true;
```



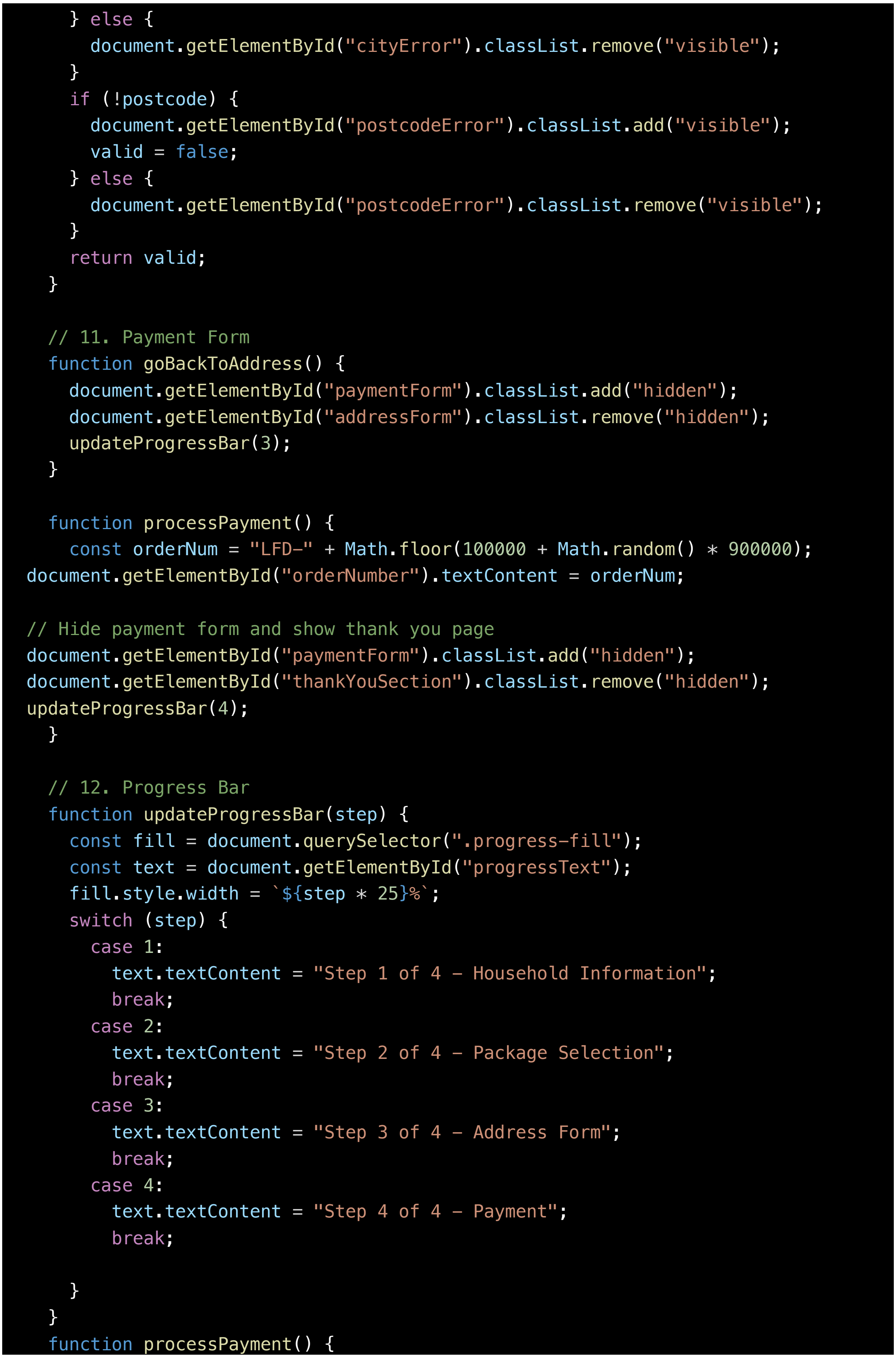
```
      ageInputs.forEach(input => {
        const age = parseInt(input.value);
        if (isNaN(age) || age < 0) {
          input.nextElementSibling.classList.add("visible");
          isValid = false;
        } else {
          input.nextElementSibling.classList.remove("visible");
        }
      });
      return isValid;
    }

    // 7. Suggest Pack
    const packData = [
      { name: "Lentils", vegan: "500g", "gluten-free": "-", vegetarian: "500g",
couple: "1kg", essentials: "500g", family: "2kg" },
      { name: "Chickpeas", vegan: "500g", "gluten-free": "-", vegetarian: "500g",
couple: "1kg", essentials: "500g", family: "2kg" },
      { name: "Tofu", vegan: "500g", "gluten-free": "-", vegetarian: "-", couple:
"-", essentials: "-", family: "-" },
      { name: "Tempeh", vegan: "500g", "gluten-free": "-", vegetarian: "-", couple:
"-", essentials: "-", family: "-" },
      { name: "Nuts & Seeds", vegan: "300g", "gluten-free": "300g", vegetarian:
"300g", couple: "500g", essentials: "300g", family: "1kg" },
      { name: "Brown Rice", vegan: "1.5kg", "gluten-free": "1.5kg", vegetarian:
"1.5kg", couple: "2kg", essentials: "1.5kg", family: "4kg" },
      { name: "Whole Wheat Pasta", vegan: "500g", "gluten-free": "-", vegetarian:
"500g", couple: "1kg", essentials: "500g", family: "2kg" },
      { name: "Oats", vegan: "1kg", "gluten-free": "1kg", vegetarian: "1kg",
couple: "1.5kg", essentials: "1kg", family: "2kg" },
      { name: "Wholegrain Bread", vegan: "1 loaf", "gluten-free": "-", vegetarian:
"1 loaf", couple: "2 loaves", essentials: "1 loaf", family: "4 loaves" },
      { name: "Potatoes", vegan: "-", "gluten-free": "2kg", vegetarian: "-",
couple: "3kg", essentials: "2kg", family: "6kg" },
      { name: "Leafy Greens", vegan: "500g", "gluten-free": "500g", vegetarian:
"500g", couple: "1kg", essentials: "500g", family: "2kg" },
      { name: "Tomatoes", vegan: "500g", "gluten-free": "500g", vegetarian: "500g",
couple: "1kg", essentials: "500g", family: "2kg" },
      { name: "Bell Peppers", vegan: "3", "gluten-free": "3", vegetarian: "3",
couple: "6", essentials: "3", family: "12" },
      { name: "Carrots", vegan: "500g", "gluten-free": "500g", vegetarian: "500g",
couple: "1kg", essentials: "500g", family: "2kg" },
      { name: "Bananas", vegan: "5", "gluten-free": "5", vegetarian: "5", couple:
"10", essentials: "5", family: "20" },
      { name: "Apples", vegan: "5", "gluten-free": "5", vegetarian: "5", couple:
"10", essentials: "5", family: "20" },
      { name: "Berries", vegan: "250g", "gluten-free": "250g", vegetarian: "250g",
couple: "500g", essentials: "250g", family: "1kg" },
      { name: "Soy Milk", vegan: "2L", "gluten-free": "-", vegetarian: "2L",
couple: "4L", essentials: "2L", family: "8L" },
      { name: "Nutritional Yeast", vegan: "100g", "gluten-free": "-", vegetarian:
"-", couple: "-", essentials: "-", family: "-" },
      { name: "Plant-based Yogurt", vegan: "500g", "gluten-free": "-", vegetarian:
"-", couple: "-", essentials: "-", family: "-" },
      { name: "Olive Oil", vegan: "500ml", "gluten-free": "500ml", vegetarian:
"500ml", couple: "1L", essentials: "500ml", family: "1L" },
      { name: "Butter", vegan: "-", "gluten-free": "-", vegetarian: "250g", couple:
"500g", essentials: "250g", family: "500g" },
      { name: "Chicken Breast", vegan: "-", "gluten-free": "1kg", vegetarian: "-",
couple: "2kg", essentials: "1kg", family: "3kg" },
      { name: "Eggs", vegan: "-", "gluten-free": "12", vegetarian: "12", couple:
"24", essentials: "12", family: "36" },
      { name: "Fish", vegan: "-", "gluten-free": "500g", vegetarian: "-", couple:
"1kg", essentials: "500g", family: "2kg" },
      { name: "Cheese", vegan: "-", "gluten-free": "250g", vegetarian: "250g",
couple: "500g", essentials: "250g", family: "1kg" },
      { name: "Greek Yogurt", vegan: "-", "gluten-free": "-", vegetarian: "500g",
couple: "1kg", essentials: "-", family: "2kg" },
      { name: "Quinoa", vegan: "-", "gluten-free": "1kg", vegetarian: "-", couple:
"2kg", essentials: "1kg", family: "4kg" },
      { name: "Gluten-Free Bread", vegan: "-", "gluten-free": "1 loaf", vegetarian:
"-", couple: "2 loaves", essentials: "1 loaf", family: "4 loaves" },
      { name: "Milk", vegan: "-", "gluten-free": "2L", vegetarian: "2L", couple:
"4L", essentials: "2L", family: "8L" }
    ];

    function suggestPack(e) {
      e.preventDefault();
      if (!validateForm()) return false;

      const householdSize =
parseInt(document.getElementById("householdSize").value);
      const dietary = document.getElementById("dietaryRequirements").value;
      let packType = "essentials";

      if (dietary === "vegan" || dietary === "vegetarian" || dietary === "gluten-
free") {
        packType = dietary;
      } else if (householdSize > 2) {
        packType = "family";
      } else if (householdSize === 2) {
        packType = "couple";
      }
      const items = calculateQuantities(packType, householdSize);
      const itemListHtml = generateItemList(items);
      document.getElementById("packItems").innerHTML = itemListHtml;
      document.getElementById("packTitle").textContent =
        packType.charAt(0).toUpperCase() + packType.slice(1) + " Pack";

      document.getElementById("householdInfoStep").classList.add("hidden");
      document.getElementById("packSuggestion").classList.remove("hidden");
```



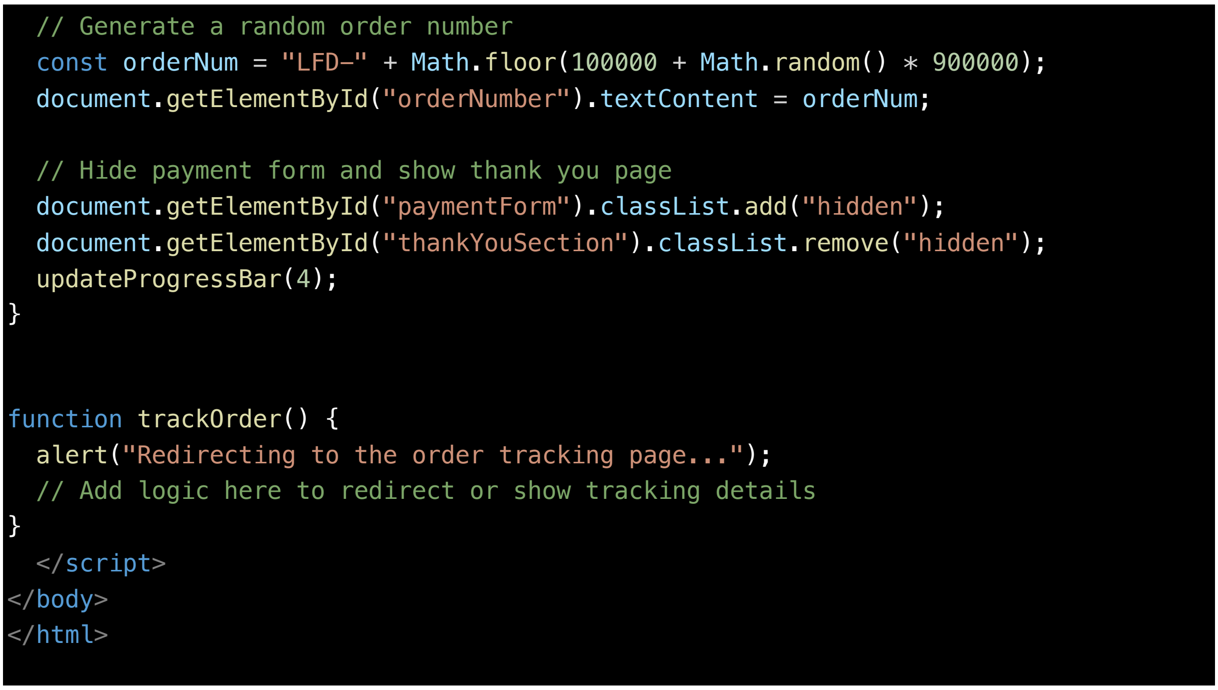
```
  updateProgressBar(2);
  return false;
}

// 8. Calculate and Generate Item List
function calculateQuantities(packType, householdSize) {
  let basePack = packType.toLowerCase();
  let multiplier = 1;

  if (["vegan", "vegetarian", "gluten-free"].includes(basePack)) {
    multiplier = householdSize;
  } else if (basePack === "family") {
    basePack = "essentials";
    multiplier = householdSize > 2 ? householdSize : 1;
  }
  return packData
    .map(item => {
      let quantity = item[basePack];
      if (quantity === "-") return null;
      const match = quantity.match(/(\d+\.?\d*)(\D+)/);
      if (!match) {
        return { name: item.name, quantity };
      }
      let [_, value, unit] = match;
      let total = parseFloat(value) * multiplier;

      // Convert grams to kg if over 1000g
      if (unit === "g" && total >= 1000) {
        total /= 1000;
        unit = "kg";
      }
      // Convert loaf/loaves as needed
      if (unit === "loaf" && total > 1) unit = "loaves";
      if (unit === "loaves" && total <= 1) unit = "loaf";

      return { name: item.name, quantity: total + unit };
    })
    .filter(Boolean);
}

function generateItemList(items) {
  return items
    .map(
      item => `
        <div class="item-row">
          <span>${item.name}</span>
          <span style="font-weight: 500;">${item.quantity}</span>
        </div>
      `
    )
    .join("");
```



```
    }

    // 9. Package Selection Confirm
    function confirmSelection() {
      document.getElementById("packSuggestion").classList.add("hidden");
      document.getElementById("addressForm").classList.remove("hidden");
      updateProgressBar(3);
    }

    function goBackToHousehold() {
      document.getElementById("packSuggestion").classList.add("hidden");
      document.getElementById("householdInfoStep").classList.remove("hidden");
      updateProgressBar(1);
    }

    // 10. Address Form Navigation
    function goBackToPackage() {
      document.getElementById("addressForm").classList.add("hidden");
      document.getElementById("packSuggestion").classList.remove("hidden");
      updateProgressBar(2);
    }

    function toPaymentStep() {
      if (!validateAddressForm()) return;
      document.getElementById("addressForm").classList.add("hidden");
      document.getElementById("paymentForm").classList.remove("hidden");
      updateProgressBar(4);
    }

    function validateAddressForm() {
      const houseNumber = document.getElementById("houseNumber").value.trim();
      const street1 = document.getElementById("streetAddress1").value.trim();
      const city = document.getElementById("city").value.trim();
      const postcode = document.getElementById("postcode").value.trim();

      let valid = true;
      if (!houseNumber) {
        document.getElementById("houseNumberError").classList.add("visible");
        valid = false;
      } else {
        document.getElementById("houseNumberError").classList.remove("visible");
      }
      if (!street1) {
        document.getElementById("streetAddress1Error").classList.add("visible");
        valid = false;
      } else {
        document.getElementById("streetAddress1Error").classList.remove("visible");
      }
      if (!city) {
        document.getElementById("cityError").classList.add("visible");
        valid = false;
```

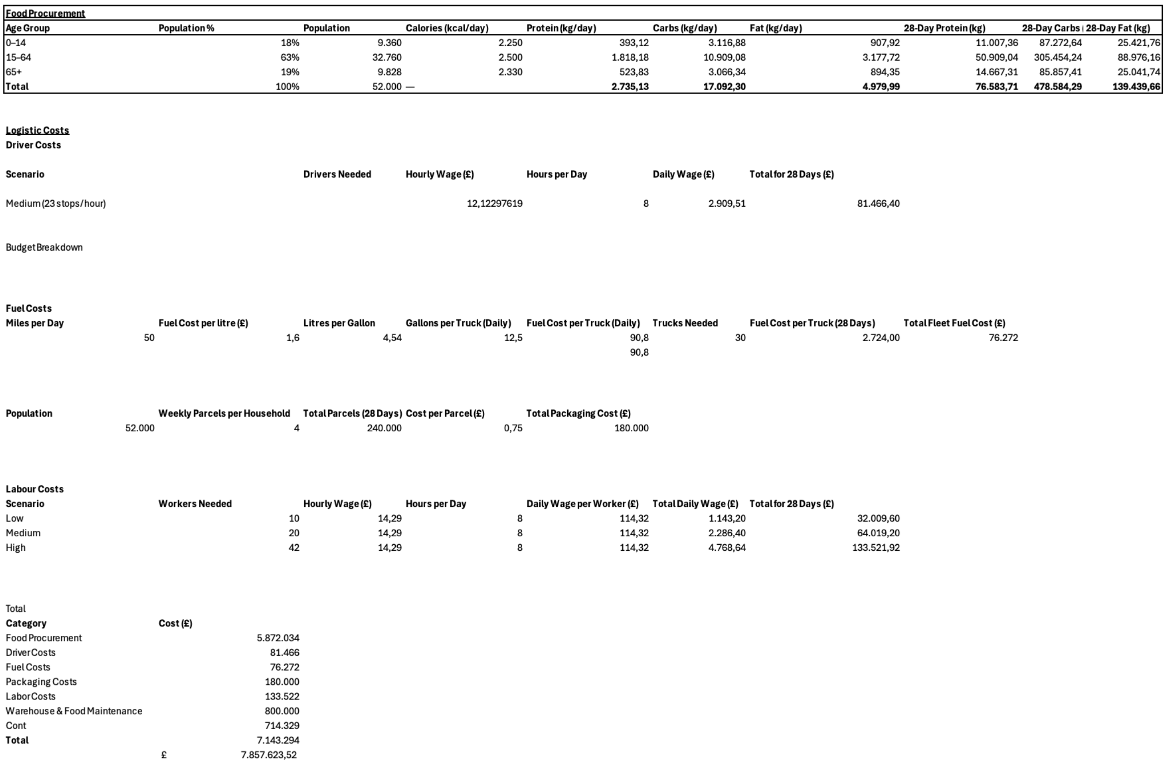


```
      } else {
        document.getElementById("cityError").classList.remove("visible");
      }
      if (!postcode) {
        document.getElementById("postcodeError").classList.add("visible");
        valid = false;
      } else {
        document.getElementById("postcodeError").classList.remove("visible");
      }
      return valid;
    }

  // 11. Payment Form
  function goBackToAddress() {
    document.getElementById("paymentForm").classList.add("hidden");
    document.getElementById("addressForm").classList.remove("hidden");
    updateProgressBar(3);
  }

  function processPayment() {
    const orderNum = "LFD-" + Math.floor(100000 + Math.random() * 900000);
document.getElementById("orderNumber").textContent = orderNum;

// Hide payment form and show thank you page
document.getElementById("paymentForm").classList.add("hidden");
document.getElementById("thankYouSection").classList.remove("hidden");
updateProgressBar(4);
  }

  // 12. Progress Bar
  function updateProgressBar(step) {
    const fill = document.querySelector(".progress-fill");
    const text = document.getElementById("progressText");
    fill.style.width = `${step * 25}%`;
    switch (step) {
      case 1:
        text.textContent = "Step 1 of 4 - Household Information";
        break;
      case 2:
        text.textContent = "Step 2 of 4 - Package Selection";
        break;
      case 3:
        text.textContent = "Step 3 of 4 - Address Form";
        break;
      case 4:
        text.textContent = "Step 4 of 4 - Payment";
        break;

    }
  }
  function processPayment() {
```



```
  // Generate a random order number
  const orderNum = "LFD-" + Math.floor(100000 + Math.random() * 900000);
  document.getElementById("orderNumber").textContent = orderNum;

  // Hide payment form and show thank you page
  document.getElementById("paymentForm").classList.add("hidden");
  document.getElementById("thankYouSection").classList.remove("hidden");
  updateProgressBar(4);
}


function trackOrder() {
  alert("Redirecting to the order tracking page...");
  // Add logic here to redirect or show tracking details
}
  </script>
</body>
</html>
```

Appendix 1.12 – Excel Sheet showing all cost calculations executed



Food Procurement

Age Group	Population %	Population	Calories (kcal/day)	Protein (kg/day)	Carbs (kg/day)	Fat (kg/day)	28-Day Protein (kg)	28-Day Carbs (	28-Day Fat (kg)
0–14	18%	9.360	2.250	393,12	3.116,88	907,92	11.007,36	87.272,64	25.421,76
15–64	63%	32.760	2.500	1.818,18	10.909,08	3.177,72	50.909,04	305.454,24	88.976,16
65+	19%	9.828	2.330	523,83	3.066,34	894,35	14.667,31	85.857,41	25.041,74
Total	100%	52.000	—	**2.735,13**	**17.092,30**	**4.979,99**	**76.583,71**	**478.584,29**	**139.439,66**

Logistic Costs

Driver Costs

Scenario	**Drivers Needed**	**Hourly Wage (£)**	**Hours per Day**	**Daily Wage (£)**	**Total for 28 Days (£)**
Medium (23 stops/hour)		12,12297619	8	2.909,51	81.466,40

Budget Breakdown

Fuel Costs

Miles per Day	**Fuel Cost per litre (£)**	**Litres per Gallon**	**Gallons per Truck (Daily)**	**Fuel Cost per Truck (Daily)**	**Trucks Needed**	**Fuel Cost per Truck (28 Days)**	**Total Fleet Fuel Cost (£)**
50	1,6	4,54	12,5	90,8	30	2.724,00	76.272
				90,8			

Population	**Weekly Parcels per Household**	**Total Parcels (28 Days)**	**Cost per Parcel (£)**	**Total Packaging Cost (£)**
52.000	4	240.000	0,75	180.000

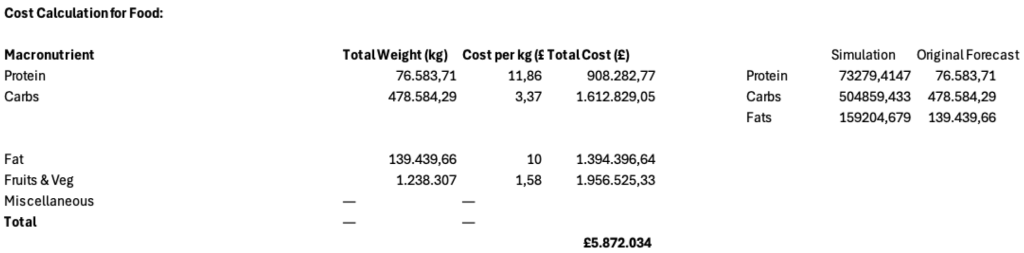
Labour Costs

Scenario	**Workers Needed**	**Hourly Wage (£)**	**Hours per Day**	**Daily Wage per Worker (£)**	**Total Daily Wage (£)**	**Total for 28 Days (£)**
Low	10	14,29	8	114,32	1.143,20	32.009,60
Medium	20	14,29	8	114,32	2.286,40	64.019,20
High	42	14,29	8	114,32	4.768,64	133.521,92

Total

Category	**Cost (£)**
Food Procurement	5.872.034
Driver Costs	81.466
Fuel Costs	76.272
Packaging Costs	180.000
Labor Costs	133.522
Warehouse & Food Maintenance	800.000
Cont	714.329
Total	7.143.294
	£ 7.857.623,52

Appendix 1.13 – Shows Precise Food Calculations, as well as Simulated food output and Original Forecasted food output.



Cost Calculation for Food:

Macronutrient	Total Weight (kg)	Cost per kg (£	Total Cost (£)
Protein	76.583,71	11,86	908.282,77
Carbs	478.584,29	3,37	1.612.829,05
Fat	139.439,66	10	1.394.396,64
Fruits & Veg	1.238.307	1,58	1.956.525,33
Miscellaneous	—	—	
Total	—	—	
			£5.872.034

	Simulation	Original Forecast
Protein	73279,4147	76.583,71
Carbs	504859,433	478.584,29
Fats	159204,679	139.439,66