TABLE OF CONTENTS

1. INTRODUCTION

IE-Tech manufactures high-quality components for robotics enthusiasts and pre-assembled robotic kits. Its product line includes aluminum frames, carbon fiber frames, manual modules, advanced control modules, and advanced sensor modules. These components are produced across three plants located in Istanbul, Ankara, and Izmir. Each has distinct production processes, resource availabilities, and capacities. Additionally, the company assembles robotic kits that integrate these components. IE-Tech must balance resource limitations, production costs, and fluctuating customer demands while maximizing profitability. The production and sales plan for a two-month horizon must be optimized.

This study provides a thorough review of Pyomo's solution to IE-Tech's production and sales optimization problem. Charts and graphs are used to assist the data-driven findings, sensitivity analysis, and graphical insights that address resource allocation, demand fulfillment, and resource allocation suggestions. The report also includes cost-benefit analyses of additional investments and strategies for managing increasing costs.

2. MAIN BODY

2.1 Assumptions

1. Since the profit is to be maximized, it is illogical to produce goods, that cannot be sold. Therefore, inventory should be 0 at the end of period 2.

2.2 Sets and Indices

i : type of plant *(i ∈ I = {1(İstanbul),2(Ankara),3(İzmir)}*

j : type of component *(j ∈ J = {1(Aluminum Frames),2(Carbon Fiber Frames) ,3(Manual Modules) ,4(Advanced Control Modules) ,5(Advanced Sensor Modules)}*

k : type of period *(k ∈ K = {1(Period 1),2(Period 2)}*

2.3 Parameters

l_{ijk} = The required labor time for producing each component j in plant i at period k (minutes /item)

la_{ik} = The available labor time for each plant i at period k (minutes)

pd_{ijk} = The required packing time for producing each component j in plant i at period k (minutes /item)

pa_{ik} = The available packing time for each plant i at period k (minutes)

ad_{ik} = The required assembly time for producing each robotic kit in plant i at period k (minutes /item)

aa_{ik} = The available assembly time for each plant i at period k (minutes)

ndc_{ijk} = The minimum demand for each component j in plant i at period k (item)

xdc_{ijk} = The maximum demand for each component j in plant i at period k (item)

ndr_{ik} = The minimum demand for robotic kit in plant i at period k (item)

xdr_{ik} = The maximum demand for robotic kit in plant i at period k (item)

pcc_{ijk} = The production cost of each component j in plant i at period k ($/item)

pcr_{ik} = The production cost of robotic kit in plant i at period k ($/item)

spc_{ijk} = The selling price of each component j in plant i at period k ($/item)

spr_{ik} = The selling price of robotic kit in plant i at period k ($/item)

r_{j} = The required amount of component j for producing each robotic kit

2.4 Decision Variables

x_{ijk} = The amount of component j produced in plant i at period k(item)

y_{ik} = The amount of robotic kit produced in plant i at period k(item)

i_{ijk} = The amount of component j in plant i inventory at the end of the period k(item)

ir_{ik} = The amount of robotic kit in plant i inventory at the end of the period k(item)

2.5 Objective Function and Constraints

$$\text{Maximize Z} = \sum_{i\in I}\sum_{j\in J}\sum_{k\in K} x_{ijk}spc_{ijk} + \sum_{i\in I}\sum_{k\in K} y_{ik}spr_{ik} - \sum_{i\in I}\sum_{j\in J}\sum_{k\in\{1\}} x_{ijk}pcc_{ijk}$$
$$- 1.12\sum_{i\in I}\sum_{j\in J}\sum_{k\in\{2\}} x_{ijk}pcc_{ijk} - \sum_{i\in I}\sum_{k\in\{1\}} y_{ik}pcr_{ik} - 1.12\sum_{i\in I}\sum_{k\in\{2\}} y_{ik}pcr_{ik}$$
$$- \sum_{i\in I}\sum_{j\in J}\sum_{k\{1\}} y_{ik}pcc_{ik}r_j - 1.12\sum_{i\in I}\sum_{j\in J}\sum_{k\in\{2\}} y_{ik}pcc_{ik}r_j$$
$$- 0.08\sum_{i\in I}\sum_{j\in J}\sum_{k\in K} i_{ijk}pcc_{ijk} - 0.08\sum_{i\in I}\sum_{k\in K} ir_{ik}pcr_{ik}$$

s.t

$\sum_{j\in J} x_{ijk}l_{ijk} \leq la_{ik}$ $\quad for\ i \in I, k \in K$ (1) Total Labor Time

$\sum_{j\in J} x_{ijk}pd_{ijk} \leq pa_{ik}$ $\quad for\ i \in I, k \in K$ (2) Total Packing Time

$y_{ik}ad_{ik} \leq aa_{ik}$ $\quad for\ i \in I, k \in K$ (3) Total Assembly Time

$\sum_{i\in I} x_{i(j=2)k} \leq 4000$ $\quad for\ k \in K$ (4) Carbon Fiber Frame Production Limit

$i_{ij(k=2)} = 0$ $\quad for\ i \in I, j \in J$ (5) Final Inventory for Components

$ir_{i(k=2)} = 0$ $\quad for\ i \in I$ (6) Final Inventory for Robotic Kits

$y_{i(k=1)} - ir_{i(k=1)} \geq ndr_{i(k=1)}$ $\quad for\ i \in I$ (7) Minimum Demand for Robotic Kits (Period 1)

$y_{i(k=2)} + ir_{i(k=1)} - ir_{i(k=2)} \geq ndr_{i(k=2)}$ $\quad for\ i \in I$ (8) Minimum Demand for Robotic Kits (Period 2)

$y_{i(k=1)} - ir_{i(k=1)} \leq xdr_{i(k=1)}$ $\quad for\ i \in I$ (9) Maximum Demand for Robotic Kits (Period 1)

$y_{i(k=2)} + ir_{i(k=1)} - ir_{i(k=2)} \leq \; xdr_{i(k=2)} \qquad for\ i \in I$ (10) Maximum Demand for Robotic Kits (Period 2)

$x_{ij(k=1)} - i_{ij(k=1)} - y_{i(k=1)}r_j \geq ndc_{ij(k=1)} \qquad for\ i \in I, \mathrm{j} \in \mathrm{J}$ (11) Minimum Demand for Components (Period 1)

$x_{ij(k=2)} + i_{ij(k=1)} - i_{ij(k=2)} - y_{i(k=2)}r_j \geq ndc_{ij(k=2)} \qquad for\ i \in I, \mathrm{j} \in \mathrm{J}$ (12) Minimum Demand for Components (Period 2)

$x_{ij(k=1)} - i_{ij(k=1)} - y_{i(k=1)}r_j \leq xdc_{ij(k=1)} \qquad for\ i \in I, \mathrm{j} \in \mathrm{J}$ (13) Maximum Demand for Components (Period 1)

$x_{ij(k=2)} + i_{ij(k=1)} - i_{ij(k=2)} - y_{i(k=2)}r_j \leq xdc_{ij(k=2)} \qquad for\ i \in I, \mathrm{j} \in \mathrm{J}$ (14) Maximum Demand for Components (Period 2)

$x_{ijk}, y_{ik}, i_{ijk}, ir_{ik} \geq 0 \qquad for\ i \in I, \mathrm{j} \in \mathrm{J}, \mathrm{k} \in \mathrm{K}$ (15)

The total labor times obtained by multiplying the labor times to be used to produce each component in each plant by the components produced must be smaller than the available labor times determined for each relevant plant (1).

The total packing times obtained by multiplying the packing times to be used to produce each component in each plant by the produced components must be smaller than the available packing times determined for each relevant plant (2).

The value obtained by multiplying the assembly times corresponding to the relevant plant and period by the number of robotic kits produced for each plant and period must be smaller than the available assembly time determined for the relevant plant and period (3).

When the amount of carbon fiber that could be obtained monthly, which is 1000 pounds (ounces?), is multiplied by the amount of carbon fiber frames that could be produced from each pound (ounce?), which is 4, the result is 4000; it must be greater than the total amount of carbon fiber frames produced monthly (4).

At the end of the 2^{nd} month, no components will remain in the inventory of any plant(5). At the end of the second month, no robotic kit will remain in the inventory of any plant (6).

When the amount of robotic kits sold in each plant in the first month is subtracted from the amount of robotic kits put into inventory for that month, the value that is obtained must be greater than or equal to the minimum demand (7) in the relevant plant and less than or equal to the maximum demand (9).

When the amount of robotic kits sold in each plant in the second month is subtracted from the amount of robotic kits put into inventory for that month, and the remaining inventory amount from the previous month is added, the value that is obtained must be greater than or equal to the minimum demand (8) in the relevant plant and less than or equal to the maximum demand (10).

When the amount of components put into inventory for that month and the amount of components used to produce robotic kits are subtracted from the amount of components sold in each plant in the first month, the value obtained must be greater than or equal to the minimum demand (11) in the relevant plant; less than or equal to the maximum demand (13).

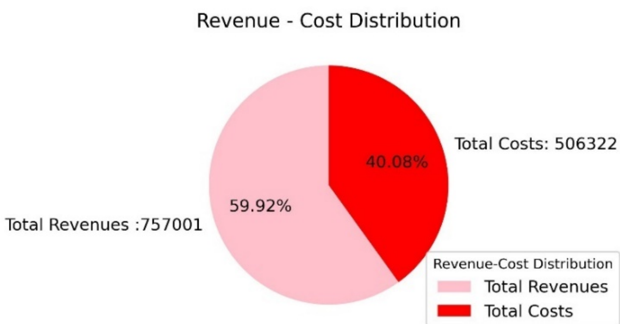
When the amount of components put into inventory for that month and the amount of components used to produce robotic kits are subtracted from the amount of components sold at each plant in the second month and the remaining inventory amount from the previous month is added, the value obtained must be greater than or equal to the minimum demand (12) in the relevant plant and less than or equal to the maximum demand (14).

All the decision variables should be nonnegative values because they express quantitative and temporal quantities (15).

2.6 Results

As a result of implementing the designed model, the maximum total profit was calculated $250,679 by subtracting total costs from total revenue (as shown in Figure 1 below).(The codes of Figure 1 in Appendix 14)

Figure 1: *Total Revenues and Total Costs*



Total revenue from selling components makes up the entirety of total revenue. However, since no robotic kits were produced and therefore sold, the total revenue from selling robotic kits is zero. (as shown in Appendix 1). While the total production cost of components constitutes the entire total cost, since no robotic kit is produced and no product is added to the inventory, other costs are zero (as shown in Appendix 2).

The execution of the model yielded the decision variable values presented in Table 1 below.

Table 1: *Components j and robotic kit produced and stored in plant I at period k*

j(component), k(period)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)	(4,1)	(4,2)	(5,1)	(5,2)
$x_{(i=1)jk}$	0	0	867.142857	100	200	200	588.571429	1100	2000	2000
$i_{(i=1)jk}$	0	0	0	0	0	0	0	0	0	0
$x_{(i=2)jk}$	0	0	1132.85714	1132.8571	200	200	30	30	2000	2000
$i_{(i=2)jk}$	0	0	0	0	0	0	0	0	0	0
$x_{(i=3)jk}$	0	0	2000	2000	100	100	370.588235	370.588235	2000	2000
$i_{(i=3)jk}$	0	0	0	0	0	0	0	0	0	0
i: type of plant (i I = {1(İstanbul),2(Ankara),3(İzmir)}										
j : type of component (j J = {1(Aluminum),2(Carbon Fiber Frame) ,3(Manual Modules) ,4(Advanced Control Modules) ,5(Advanced Sensor Modules)} k : type of period (k K = {1(Period 1),2(Period 2)}										
i(plant), k(period)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)				
y_{ik}	0	0	0	0	0	0				
ir_{ik}	0	0	0	0	0	0				

Reduced cost indicates the additional cost required to include an unproduced product into production. As seen in Table 1, aluminum frame was not produced in Istanbul at period 1. If the aluminum frame were to be produced in Istanbul in period 1, there would have been a decrease of $1.33 per item in maximum total profit.

The maximum total profit decreases per item required for the production and store of components in Istanbul, Ankara, and Izmir are shown in Table 2. The reason for the values of 0 in the x_{ijk} row of tables is the assumption that the relevant component has already been produced. The reason for the values of 0 in the i_{ijk} rows is the assumption that there is no inventory left for the second month.

Table 2: *Reduced Cost for Variable x_{ijk} and i_{ijk} of component j in plant i at period k*

j(component),k(period)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)	(4,1)	(4,2)	(5,1)	(5,2)
$x_{(i=1)jk}$	-1.333333	-1.253333	0	0	0	0	0	0	0	0
$i_{(i=1)jk}$	-0.48	0	-0.706667	0	-0.84	0	-0.8	0	-0.16	0
$x_{(i=2)jk}$	-1.333333	-0.44	0	0	0	0	0	0	0	0
$i_{(i=2)jk}$	-0.4	0	-1.44	0	-1.72	0	-1.48	0	-1.1733333	0
$x_{(i=3)jk}$	-3.823529	-3.392941	0	0	0	0	0	0		0
$i_{(i=1)jk}$	-0.56	0	-1.137255	0	-1.24	0	-0.96	0	-0.2752941	0
i: type of plant (i I = {1(İstanbul),2(Ankara),3(İzmir)}										
j : type of component (j J = {1(Aluminum),2(Carbon Fiber Frame) ,3(Manual Modules) ,4(Advanced Control Modules) ,5(Advanced Sensor Modules)} k : type of period (k K = {1(Period 1),2(Period 2)}										

If the components are to continue being produced at the same levels, their profit per item values must be in the ranges specified in Tables 3 and 4. For example, the selling price of a carbon fiber frame produced in Istanbul in the first period is \$25 and its production cost is \$19. Since no inventory is used, when the production cost is subtracted from the selling price, the profit per item is obtained \$6. If this value remains between 5.33 and 6.50, the production quantity of the carbon fiber frame and other components will not change.

Table 3: *The lowest profit per item values required to maintain the same production levels of variable x_{ijk}*

j(component),k(period)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)	(4,1)	(4,2)	(5,1)	(5,2)
İstanbul	-Inf	-Inf	5.33333	3.01333	-Inf	2.68000	7.84000	5.58000	8.00000	10.72000
Ankara	-Inf	-Inf	6.17867	4.41778	-Inf	.68000	-Inf	4.20000	9.04762	11.94667
İzmir	-Inf	-Inf	9.49020	7.55294	-Inf	3.16000	9.70750	4.71467	9.41176	14.48471
j : type of component (j J = {1(Aluminum),2(Carbon Fiber Frame) ,3(Manual Modules) ,4(Advanced Control Modules) ,5(Advanced Sensor Modules)} k : type of period (k K = {1(Period 1),2(Period 2)}										

Table 4: The highest profit per item values required to maintain the same production levels of variable x_{ijk}

j(component),k(period)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)	(4,1)	(4,2)	(5,1)	(5,2)
İstanbul	5.33333	4.53333	6.50980	4.53333	4.84000	5.66667	9.00000	6.96000	14.16000	+Inf
Ankara	6.33333	4.84000	11.86667	5.66133	4.72000	4.72000	8.14286	6.22286	17.17333	+Inf
İzmir	8.82353	7.55294	11.13725	+Inf	6.24000	8.56000	10.57778	8.61333	18.27529	+Inf
j : type of component (j J = {1(Aluminum),2(Carbon Fiber Frame) ,3(Manual Modules) ,4(Advanced Control Modules) ,5(Advanced Sensor Modules)} k : type of period (k K = {1(Period 1),2(Period 2)}										

The component production chart by plant is given in Figure 2. The reason why more components are produced in Izmir is that the profit per item values are higher in Izmir than in other plants.

Figure 2: *Distribution of Product according to the Plants*

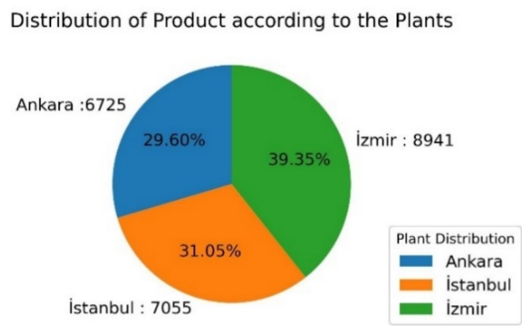
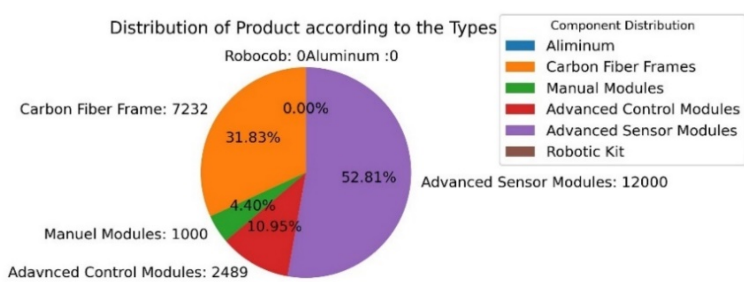


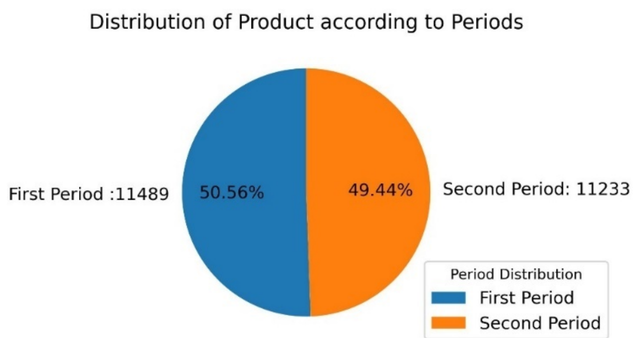
Figure 3 shows the production graph according to product types. The reason why the Advanced Sensor Module is produced more than other components is because its profit per item value is high. The reason why the Aluminum Frame is not produced at all is because it has less profit per item compared to the labor and packing time it uses. The reason why the Robotic Kit is not produced is because the total production cost exceeds the sales price.

Figure 3: *Distribution of Product according to the Types*



The component production chart by periods is given in Figure 4. Due to the 12% increase in production cost in Period 2, more production occurs in Period 1. (The python code of Figure2-Figure3 – Figure4 can be shown in Appendix 15)

Figure 4: *Distribution of Product according to the Periods*



The first row of Table 5 shows the profit amounts must be given up to produce one robotic kit. The second row shows the profit amounts must be given up to keep one robotic kit in inventory.

Table 5: *Reduced Cost Table for Variable y and ir*

i(plant),k(period)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)
y_{ik}	-361	-439.12	-339	-414.48	-388	-471.76
ir_{ik}	-14.24	0	-14	0	-14.4	0
i : type of plant (i I = {1(İstanbul),2(Ankara),3(İzmir)} k : type of period (k K = {1(Period 1),2(Period 2)}						

The shadow price indicates how a one-unit increase in a constraint would impact the maximum total profit. For instance, as shown in Table 6, if the labor availability of components (constraint 1) in Ankara at period 1 was 15,001 rather than 15,000 minutes, the maximum total profit would increase by $1.80. Similarly, if the packing availability of components (constraint 2) in Ankara at period 1 was 40,001 rather than 40,000 minutes, the profit would remain the same.

Table 6: *Shadow Prices of Constraints 1,2 and 3*

i(plant),k(period)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)
Constraint (1)	0	0	1.80952381	1.38285714	0	0
Constraint (2)	1.333333333	1.13333333	0	0	1.1764706	1.00705882
Constraint (3)	0	0	0	0	0	0
i : type of plant (i I = {1(İstanbul),2(Ankara),3(İzmir)} k : type of period (k K = {1(Period 1),2(Period 2)}						

Likewise, as shown in Table 7, if the minimum demand of manual module (constraint 11) in Istanbul at period 1 had been 201 instead of 200, the maximum total profit would have decreased by \$2.66. The constraints regarding the usage of these resources can be found in Appendix 3 and Appendix 4.

Table 7: *Shadow Prices of Constraints 5,11,12,13 and 14*

i(plant),j(component)	(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(3,1)	(3,2)	(3,3)	(3,4)	(3,5)
Constraint (5)	-0.48	-2.33333333	-2.46666667	-0.8	2	-0.4	-1.44	-4.22286	-1.42286	4.285714	-0.56	-1.55294	-4.56	-0.96	4.543529
Constraint (11)	0	0	-2.66666667	0	0	0	0	-5.14286	-1.14286	0	0	0	-5	0	0
Constraint (12)	0	-0.81333333	-2.14666667	0	0	0	0	-3.82286	-0.54286	0	0	0	-4.16	0	0
Constraint (13)	0	0	0	0	6	0	0	0	0	6.952381	0	0.509804	0	0	8.588235
Constraint (14)	0	0	0	0	4.08	0	0	0	0	6.205714	0	0.047059	0	0	6.703529
i : type of plant (i I = {1(Istanbul),2(Ankara),3(Izmir)} j : type of component (j J = {1(Aluminum),2(Carbon Fiber Frame) ,3(Manual Modules) ,4(Advanced Control Modules) ,5(Advanced Sensor Modules)}															

The slack value indicates how much less than a certain value the resource is used when the resource usage should be below a certain value. In Table 8, labor time availability (1) in Istanbul at period 1, there are 633.5 minutes that are not used. Slack is meaningless when the resource usage must be above a certain value. Therefore, it is shown as inf in the table. (Values of other constraints are shown in Appendices 5-6-7).

Table 8: *Slack value of Constraints 1,2 and 3*

i(plant),k(period)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)
Constraint (1)	633.5714286	250	0.00	0.00	1932.3529	1932.35294
Constraint (2)	0	0	16200	16200	0	0
Constraint (3)	5500	5500	5000	5000	6000	6000
i : type of plant (i I = {1(İstanbul),2(Ankara),3(İzmir)} k : type of period (k K = {1(Period 1),2(Period 2)}						

The excess value indicates how much more resources are used than a certain value when the resource usage should be above a certain value. For example, minimum demand in period 1 in Istanbul, advanced sensor modules produced 1900 units more than the minimum demand. In (13) the resource usage should be below a certain value. Therefore, excess values are meaningless and are expressed with inf. (Values of other constraints can be found in Appendices 8-9-10)

Table 9: *Excess value of Constraints 11,12,13 and 14*

i(plant),j(component)	(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(3,1)	(3,2)	(3,3)	(3,4)	(3,5)
Constraint (11)	0	767.1428571	0	558.571429	1900	0	1032.857	0	0	1900	0	1950	0	355.5882	1900
Constraint (12)	0	0	0	1070	1900	0	1032.857	0	0	1900	0	1950	0	355.5882	1900
Constraint (13)	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf
Constraint (14)	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf
i : type of plant (i I = {1(Istanbul),2(Ankara),3(Izmir)} j : type of component (j J = {1(Aluminum),2(Carbon Fiber Frame) ,3(Manual Modules) ,4(Advanced Control Modules) ,5(Advanced Sensor Modules)}															

In Table 10, the "Activity" column tells how much carbon fiber is used. The "Activity Range" column tells us what the carbon fiber limit would be if we were to keep the same type of source we produce but change the production amount. The first row gives the lower limit, the second row gives the upper limit. The "Obj value at break point variable" column tells us what the maximum profit value would be if the production were at the lower and upper limits of the carbon fiber constraint. (Other data can be found in Appendix 11-12-13)

Table 10: *The Effect of Constraint 4 to Maximum Profit*

Row name	Activity	Slack/Marginal	Lower bound/Upper bound	Activity range	Obj.Coeff range	Obj value at break point variable
Constraint 4 at period 1	4000.00000	.	-Inf	3232.85714	-.66667	250167.54622
		.66667	4000.00000	4837.85714	+Inf	251237.54622
Constraint 4 at period 2	3232.85714	767.14286	-Inf	1386.19048	-.04706	250526.84034
		.	4000.00000	4837.85714	.81333	253308.36527

Discussions

The key managerial insights are derived from the results of the optimization model in terms of the production and sales strategy of IE-Tech. The focus of this study was to improve the production and storage of robotic kits and components across locations in Istanbul, Ankara, and Izmir, with a view to maximizing profitability while satisfying a number of constraints. This study showed that a total profit of $250,679 was made just from the creation and sale of components. As the total cost of production was higher than the selling price, no robotic kit was produced. This would mean that for this product to be competitive, either the manufacturing cost of robotic kits should be optimized or the price strategy reviewed. The Advanced Sensor Module was produced the most since it had the highest profit margin, while products like the Robotic Kits and Aluminum Frame had to be excluded from production because of their low cost-to-price ratio. As shown in the reduced cost analysis in Table 2, less profitable products were excluded from the production plan.

Izmir was the most effective plant and contributed the most to profit as it produces at the highest volumes. This would be due to İzmir's profitability per item being larger than İstanbul's and Ankara's. Furthermore, timing was crucial to avoid the 12% increase in cost in Period 2, most of the production occurred during the first period. Shadow price analysis identified the labor and packaging bounds to be important, especially in Ankara. A moderate increase in the supply of laborers in Ankara can bring high profits.

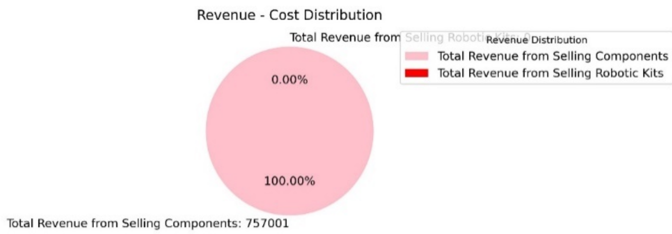
Conclusion and Recommendations

Results underline how important the correct resource allocation and strategic production planning are for optimal profitability. Simplification of the product portfolio is recommended by focusing resources on high-margin products, such as the Advanced Sensor Module, while revisiting the feasibility of using robotic kits and aluminum frames for further optimization of performance. Moreover, their profitability can be improved by reordering their price or reducing the costs of producing the items in question. While re-evaluating the production procedures for low-margin products, IE-Tech should continue to prioritize the components with high profit per item. Further profitability can be realized by investment in Izmir's high profit operations, but ways of achieving greater revenue efficiency in Istanbul and Ankara need to be explored. To exploit the shadow price information, such as the potential for increased labor supply in Ankara, which has proved promising for profits, dynamic resource allocation mechanisms should be implemented. With the existing advantage of İzmir in profit per item, investment in its production capacity may yield higher profits. Future production plans should be guided by period-specific production methods emphasizing lower-cost periods to ensure cost efficiency. The increase in cost in period 2 underlines the need for front-loading of production wherever possible. Plans for future production should continue to take advantage of cost benefits from past eras.

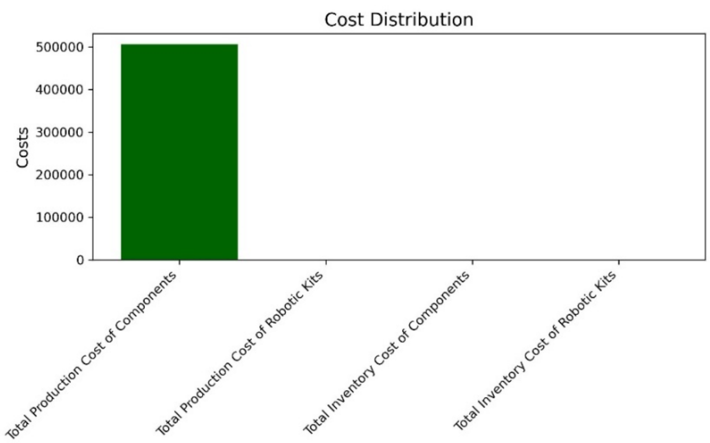
In conclusion, the overall profitability will increase significantly due to cost-effective facilities, prioritization of profitable items, and modification in resource allocation. These recommendations provide practical information that the decision-makers may utilize for improving tactics in production and ensuring long-term success.

APPENDIX

Appendix 1. Revenue Distribution



Appendix 2. Cost Distribution



Appendix 3. Shadow Prices of Constraints 7, 8, 9 and 10

plants	İstanbul	Ankara	İzmir
Constraint (7)	0	0	0
Constraint (8)	0	0	0
Constraint (9)	0	0	0
Constraint (10)	0	0	0

Appendix 4. Shadow Price of Constraint 4

period	Period 1	Period 2
Constraint (4)	0.666666667	0

Appendix 5. Slack Value of Constraint 4

period	Period 1	Period 2
Constraint (4)	0.00	767.1428571

Appendix 6. Slack value of Constraints 11,12,13 and 14

i(plant),j(component)	(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(3,1)	(3,2)	(3,3)	(3,4)	(3,5)
Constraint (11)	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf
Constraint (12)	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf
Constraint (13)	2000	1132.857143	1800	1411.42857	0	2000	867.1429	1800	1970	0	2000	0	1900	1629.412	0
Constraint (14)	2000	1900	1800	900	0	2000	867.1429	1800	1970	0	2000	0	1900	1629.412	0
i : type of plant (i I = {1(Istanbul),2(Ankara),3(Izmir)} j : type of component (j J = {1(Aluminum),2(Carbon Fiber Frame) ,3(Manual Modules) ,4(Advanced Control Modules) ,5(Advanced Sensor Modules)}															

Appendix 7. Slack value of Constraints 6,7,8,9 and 10

plants	İstanbul	Ankara	İzmir
Constraint (7)	inf	inf	inf
Constraint (8)	inf	inf	inf
Constraint (9)	200	200	200
Constraint (10)	200	200	200

Appendix 8. Excess value of Constraints 1,2 and 3

i(plant),k(period)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)
Constraint (1)	inf	inf	inf	inf	inf	inf
Constraint (2)	inf	inf	inf	inf	inf	inf
Constraint (3)	inf	inf	inf	inf	inf	inf
i : type of plant (i I = {1(İstanbul),2(Ankara),3(İzmir)} k : type of period (k K = {1(Period 1),2(Period 2)}						

Appendix 9. Excess value of Constraints 6,7,8,9 and 10

plants	İstanbul	Ankara	İzmir
Constraint (6)	0	0	0
Constraint (7)	0	0	0
Constraint (8)	0	0	0
Constraint (9)	inf	inf	inf
Constraint (10)	inf	inf	inf

Appendix 10. Excess value of Constraint 4

period	Period 1	Period 2
Constraint (4)	inf	inf

Appendix 11. Sensitivity Analysis between Constraints (1) and (5)

c_u_eLabCons(Istanbul_1)_	BS	11366.428	633.57143	-Inf	11166.42857	-.84000	241131.17479
			.	12000.00000	11925.00000	.16000	252497.60336
c_u_eLabCons(Istanbul_2)_	BS	11750.000	250.00000	-Inf	11191.42857	-.16000	248798.97479
			.	12000.00000	11950.00000	.84000	260548.97479
c_u_eLabCons(Ankara_1)_	NU	15000.000	.	-Inf	12067.50000	-1.80952	245372.54622
			1.80952	15000.00000	17685.00000	+Inf	255537.54622
c_u_eLabCons(Ankara_2)_	NU	15000.000	.	-Inf	11385.00000	-1.38286	245679.94622
			1.38286	15000.00000	17685.00000	+Inf	254391.94622
c_u_eLabCons(Izmir_1)_	BS	20067.647	1932.35294	-Inf	18467.50000	-2.22222	206084.20355
			.	22000.00000	20545.46875	.21767	255047.18820
c_u_eLabCons(Izmir_2)_	BS	20067.647	1932.35294	-Inf	19589.82537	-.21767	246310.76138
			.	22000.00000	20936.66667	.10000	252685.73950
c_u_ePacCons(Istanbul_1)_	NU	20000.000	.	-Inf	16648.57143	-1.33333	246210.40336
			1.33333	20000.00000	21267.14286	+Inf	252368.49860
c_u_ePacCons(Istanbul_2)_	NU	20000.000	.	-Inf	13580.00000	-1.13333	243402.97479
			1.13333	20000.00000	20500.00000	+Inf	251245.64146
c_u_ePacCons(Ankara_1)_	BS	23800.000	16200.00000	-Inf	22189.00000	-.39111	241370.53035
			.	40000.00000	25411.00000	2.31746	305834.53035
c_u_ePacCons(Ankara_2)_	BS	23800.000	16200.00000	-Inf	16570.00000	-.69143	234222.97479
			.	40000.00000	25411.00000	.39111	259987.41923
c_u_ePacCons(Izmir_1)_	NU	35000.000	.	-Inf	31977.50000	-1.17647	247123.09244
			1.17647	35000.00000	38650.00000	+Inf	254973.09244
c_u_ePacCons(Izmir_2)_	NU	35000.000	.	-Inf	31977.50000	-1.00706	247635.13950
			1.00706	35000.00000	38650.00000	+Inf	254354.73950
c_u_AssCons(Istanbul_1)_	BS	.	5500.00000	-Inf	.	-Inf	250678.97479
			.	5500.00000	13000.00000	5.55385	250678.97479
c_u_AssCons(Istanbul_2)_	BS	.	5500.00000	-Inf	.	-Inf	250678.97479
			.	5500.00000	13000.00000	6.75569	250678.97479
c_u_AssCons(Ankara_1)_	BS	.	5000.00000	-Inf	.	-Inf	250678.97479
			.	5000.00000	12000.00000	5.65000	250678.97479
c_u_AssCons(Ankara_2)_	BS	.	5000.00000	-Inf	.	-Inf	250678.97479
			.	5000.00000	12000.00000	6.90800	250678.97479
c_u_AssCons(Izmir_1)_	BS	.	6000.00000	-Inf	.	-Inf	250678.97479
			.	6000.00000	13000.00000	5.96923	250678.97479
c_u_AssCons(Izmir_2)_	BS	.	6000.00000	-Inf	.	-Inf	250678.97479
			.	6000.00000	13000.00000	7.25785	250678.97479
c_u_CfCons(1)_	NU	4000.0000	.	-Inf	3232.85714	-.66667	250167.54622
			.66667	4000.00000	4837.85714	+Inf	251237.54622
c_u_CfCons(2)_	BS	3232.8571	767.14286	-Inf	1386.19048	-.04706	250526.84034
			.	4000.00000	4837.85714	.81333	253308.36527
c_e_eCompInv(Istanbul_Aliminum)_	NS	.	.	.	.	-Inf	250678.97479
			-.48000	.	.	+Inf	250678.97479
c_e_eCompInv(Istanbul_CarbonFibe	NS	.	.	.	.	-Inf	250678.97479
			-2.33333	.	767.14286	+Inf	248888.97479
c_e_eCompInv(Istanbul_Manual)_	NS	.	.	.	.	-Inf	250678.97479
			-2.46667	.	1284.00000	+Inf	247511.77479
c_e_eCompInv(Istanbul_Control)_	NS	.	.	.	.	-Inf	250678.97479
			-.80000	.	1070.00000	+Inf	249822.97479
c_e_eCompInv(Istanbul_Sensor)_	NS	.	.	.	.	-Inf	250678.97479
			2.00000	.	250.00000	+Inf	251178.97479
c_e_eCompInv(Ankara_Aliminum)_	NS	.	.	.	.	-Inf	250678.97479
			-.40000	.	.	+Inf	250678.97479
c_e_eCompInv(Ankara_CarbonFiber	NS	.	.	.	.	-Inf	250678.97479
			-1.44000	.	1032.85714	+Inf	249191.66050
c_e_eCompInv(Ankara_Manual)_	NS	.	.	.	.	-Inf	250678.97479
			-4.22286	.	803.33333	+Inf	247286.61289
c_e_eCompInv(Ankara_Control)_	NS	.	.	.	.	-Inf	250678.97479
			-1.42286	.	803.33333	+Inf	249535.94622
c_e_eCompInv(Ankara_Sensor)_	NS	.	.	.	.	-Inf	250678.97479
			4.28571	.	723.00000	+Inf	253777.54622
c_e_eCompInv(Izmir_Aliminum)_	NS	.	.	.	.	-Inf	250678.97479
			-.56000	.	.	+Inf	250678.97479
c_e_eCompInv(Izmir_CarbonFiber)_	NS	.	.	.	.	-Inf	250678.97479
			-1.55294	.	403.00000	+Inf	250053.13950
c_e_eCompInv(Izmir_Manual)_	NS	.	.	.	.	-Inf	250678.97479
			-4.56000	.	355.58824	+Inf	249057.49244
c_e_eCompInv(Izmir_Control)_	NS	.	.	.	.	-Inf	250678.97479
			-.96000	.	355.58824	+Inf	250337.61008
c_e_eCompInv(Izmir_Sensor)_	NS	.	.	.	.	-Inf	250678.97479
			4.54353	.	377.81250	+Inf	252395.57700

Appendix 12. Sensitivity Analysis between Constraints (6) and (12)

c_e_eRKInv(Istanbul)_	NS	.	.	.	.	-Inf	250678.97479
			-14.24000	.	.	+Inf	250678.97479
c_e_eRKInv(Ankara)_	NS	.	.	.	.	-Inf	250678.97479
			-14.00000	.	.	+Inf	250678.97479
c_e_eRKInv(Izmir)_	NS	.	.	.	.	-Inf	250678.97479
			-14.40000	.	.	+Inf	250678.97479
c_l_eMRK(Istanbul)_	BS	.	.	.	.	-Inf	250678.97479
			.	+Inf	200.00000	14.24000	250678.97479
c_l_eMRK(Ankara)_	BS	.	.	.	.	-Inf	250678.97479
			.	+Inf	200.00000	14.00000	250678.97479
c_l_eMRK(Izmir)_	BS	.	.	.	.	-Inf	250678.97479
			.	+Inf	200.00000	14.40000	250678.97479
c_u_eMaxRK(Istanbul_1)_	BS	.	200.00000	-Inf	-200.00000	-14.24000	250678.97479
			.	200.00000	84.61538	361.00000	250678.97479
c_u_eMaxRK(Istanbul_2)_	BS	.	200.00000	-Inf	.	-Inf	250678.97479
			.	200.00000	+Inf	14.24000	250678.97479
c_u_eMaxRK(Ankara_1)_	BS	.	200.00000	-Inf	-200.00000	-14.00000	250678.97479
			.	200.00000	83.33333	339.00000	250678.97479
c_u_eMaxRK(Ankara_2)_	BS	.	200.00000	-Inf	.	-Inf	250678.97479
			.	200.00000	+Inf	14.00000	250678.97479
c_u_eMaxRK(Izmir_1)_	BS	.	200.00000	-Inf	-200.00000	-14.40000	250678.97479
			.	200.00000	92.30769	388.00000	250678.97479
c_u_eMaxRK(Izmir_2)_	BS	.	200.00000	-Inf	.	-Inf	250678.97479
			.	200.00000	+Inf	14.40000	250678.97479
c_l_eMDC1(Istanbul_Aliminum)_	BS	.	.	.	-2000.00000	-.48000	250678.97479
			.	+Inf	837.85714	1.33333	250678.97479
c_l_eMDC1(Istanbul_CarbonFiber)_	BS	867.14286	-767.14286	100.00000	.	-.66667	250100.87955
			.	+Inf	1705.00000	.50980	251121.04762
c_l_eMDC1(Istanbul_Manual)_	NL	200.00000	.	200.00000	.	-Inf	251212.30812
			-2.66667	+Inf	870.28571	2.66667	248891.54622
c_l_eMDC1(Istanbul_Control)_	BS	588.57143	-558.57143	30.00000	.	-.16000	250584.80336
			.	+Inf	1100.00000	1.00000	251267.54622
c_l_eMDC1(Istanbul_Sensor)_	BS	2000.0000	-1900.00000	100.00000	588.57143	-6.00000	238678.97479
			.	+Inf	2000.00000	+Inf	+Inf
c_l_eMDC1(Ankara_Aliminum)_	BS	.	.	.	-2000.00000	-.40000	250678.97479
			.	+Inf	837.85714	1.33333	250678.97479
c_l_eMDC1(Ankara_CarbonFiber)_	BS	1132.8571	-1032.85714	100.00000	365.71429	-.82133	249748.52146
			.	+Inf	1900.00000	4.86667	256192.21289
c_l_eMDC1(Ankara_Manual)_	NL	200.00000	.	200.00000	.	-Inf	251707.54622
			-5.14286	+Inf	851.66667	5.14286	247327.54622
c_l_eMDC1(Ankara_Control)_	NL	30.00000	.	30.00000	.	-Inf	250713.26050
			-1.14286	+Inf	681.66667	1.14286	249934.21289
c_l_eMDC1(Ankara_Sensor)_	BS	2000.0000	-1900.00000	100.00000	1463.00000	-6.95238	236774.21289
			.	+Inf	2000.00000	+Inf	+Inf
c_l_eMDC1(Izmir_Aliminum)_	BS	.	.	.	-2000.00000	-.56000	250678.97479
			.	+Inf	403.00000	3.82353	250678.97479
c_l_eMDC1(Izmir_CarbonFiber)_	BS	2000.0000	-1950.00000	50.00000	1162.14286	-.50980	249659.36695
			.	+Inf	2000.00000	+Inf	+Inf
c_l_eMDC1(Izmir_Manual)_	NL	100.00000	.	100.00000	.	-Inf	251178.97479
			-5.00000	+Inf	455.58824	5.00000	248901.03361
c_l_eMDC1(Izmir_Control)_	BS	370.58824	-355.58824	15.00000	.	-.29250	250570.57773
			.	+Inf	1109.87395	.57778	250893.09244
c_l_eMDC1(Izmir_Sensor)_	BS	2000.0000	-1900.00000	100.00000	268.75000	-8.58824	233502.50420
			.	+Inf	2000.00000	+Inf	+Inf
c_l_eMDC2(Istanbul_Aliminum)_	BS	.	.	.	.	-Inf	250678.97479
			.	+Inf	.	.48000	250678.97479
c_l_eMDC2(Istanbul_CarbonFiber)_	NL	100.00000	.	100.00000	.	-Inf	250760.30812
			-.81333	+Inf	867.14286	.81333	250055.03193
c_l_eMDC2(Istanbul_Manual)_	NL	200.00000	.	200.00000	.	-Inf	251108.30812
			-2.14667	+Inf	1484.00000	2.14667	247922.65479
c_l_eMDC2(Istanbul_Control)_	BS	1100.0000	-1070.00000	30.00000	588.57143	-1.22000	249336.97479
			.	+Inf	1658.57143	.16000	250854.97479
c_l_eMDC2(Istanbul_Sensor)_	BS	2000.0000	-1900.00000	100.00000	1100.00000	-4.08000	242518.97479
			.	+Inf	2000.00000	+Inf	+Inf
c_l_eMDC2(Ankara_Aliminum)_	BS	.	.	.	.	-Inf	250678.97479
			.	+Inf	.	.40000	250678.97479
c_l_eMDC2(Ankara_CarbonFiber)_	BS	1132.8571	-1032.85714	100.00000	.	-.42222	250200.65733
			.	+Inf	1900.00000	.82133	251609.42812
c_l_eMDC2(Ankara_Manual)_	NL	200.00000	.	200.00000	.	-Inf	251443.54622
			-3.82286	+Inf	1003.33333	3.82286	247607.94622
c_l_eMDC2(Ankara_Control)_	NL	30.00000	.	30.00000	.	-Inf	250695.26050
			-.54286	+Inf	833.33333	.54286	250242.87955
c_l_eMDC2(Ankara_Sensor)_	BS	2000.0000	-1900.00000	100.00000	1463.00000	-6.20571	238267.54622
			.	+Inf	2000.00000	+Inf	+Inf
c_l_eMDC2(Izmir_Aliminum)_	BS	.	.	.	.	-Inf	250678.97479
			.	+Inf	.	.56000	250678.97479
c_l_eMDC2(Izmir_CarbonFiber)_	BS	2000.0000	-1950.00000	50.00000	153.33333	-.04706	250584.85714
			.	+Inf	2000.00000	+Inf	+Inf
c_l_eMDC2(Izmir_Manual)_	NL	100.00000	.	100.00000	.	-Inf	251094.97479
			-4.16000	+Inf	455.58824	4.16000	249199.72773
c_l_eMDC2(Izmir_Control)_	BS	370.58824	-355.58824	15.00000	.	-3.84533	249253.93950
			.	+Inf	2000.00000	.05333	250698.73950
c_l_eMDC2(Izmir_Sensor)_	BS	2000.0000	-1900.00000	100.00000	268.75000	-6.70353	237271.91597
			.	+Inf	2000.00000	+Inf	+Inf

Appendix 13. Sensitivity Analysis of Constraints (13) and (14)

c_u_eMaxDC1(Istanbul_Aliminum)_	BS	.	2000.00000	-Inf	.	-.48000	250678.97479
			.	2000.00000	837.85714	1.33333	250678.97479
c_u_eMaxDC1(Istanbul_CarbonFiber	BS	867.14286	1132.85714	-Inf	100.00000	-.66667	250100.87955
			.	2000.00000	1705.00000	.50980	251121.04762
c_u_eMaxDC1(Istanbul_Manual)_	BS	200.00000	1800.00000	-Inf	200.00000	-Inf	-Inf
			.	2000.00000	870.28571	2.66667	251212.30812
c_u_eMaxDC1(Istanbul_Control)_	BS	588.57143	1411.42857	-Inf	30.00000	-.16000	250584.80336
			.	2000.00000	1100.00000	1.00000	251267.54622
c_u_eMaxDC1(Istanbul_Sensor)_	NU	2000.0000	.	-Inf	588.57143	-6.00000	242210.40336
			6.00000	2000.00000	2558.57143	+Inf	254030.40336
c_u_eMaxDC1(Ankara_Aliminum)_	BS	.	2000.00000	-Inf	.	-.40000	250678.97479
			.	2000.00000	837.85714	1.33333	250678.97479
c_u_eMaxDC1(Ankara_CarbonFiber)_	BS	1132.8571	867.14286	-Inf	365.71429	-.82133	249748.52146
			.	2000.00000	1900.00000	4.86667	256192.21289
c_u_eMaxDC1(Ankara_Manual)_	BS	200.00000	1800.00000	-Inf	200.00000	-Inf	-Inf
			.	2000.00000	851.66667	5.14286	251707.54622
c_u_eMaxDC1(Ankara_Control)_	BS	30.00000	1970.00000	-Inf	30.00000	-Inf	-Inf
			.	2000.00000	681.66667	1.14286	250713.26050
c_u_eMaxDC1(Ankara_Sensor)_	NU	2000.0000	.	-Inf	1463.00000	-6.95238	246945.54622
			6.95238	2000.00000	2586.50000	+Inf	254756.54622
c_u_eMaxDC1(Izmir_Aliminum)_	BS	.	2000.00000	-Inf	.	-.56000	250678.97479
			.	2000.00000	403.00000	3.82353	250678.97479
c_u_eMaxDC1(Izmir_CarbonFiber)_	NU	2000.0000	.	-Inf	1162.14286	-.50980	250251.83193
			.50980	2000.00000	2403.00000	+Inf	250884.42577
c_u_eMaxDC1(Izmir_Manual)_	BS	100.00000	1900.00000	-Inf	100.00000	-Inf	-Inf
			.	2000.00000	455.58824	5.00000	251178.97479
c_u_eMaxDC1(Izmir_Control)_	BS	370.58824	1629.41176	-Inf	15.00000	-.29250	250570.57773
			.	2000.00000	1109.87395	.57778	250893.09244
c_u_eMaxDC1(Izmir_Sensor)_	NU	2000.0000	.	-Inf	268.75000	-8.58824	235810.59244
			8.58824	2000.00000	2377.81250	+Inf	253923.71744
c_u_eMaxDC2(Istanbul_Aliminum)_	BS	.	2000.00000	-Inf	.	-Inf	250678.97479
			.	2000.00000	.	.48000	250678.97479
c_u_eMaxDC2(Istanbul_CarbonFiber	BS	100.00000	1900.00000	-Inf	100.00000	-Inf	-Inf
			.	2000.00000	867.14286	.81333	250760.30812
c_u_eMaxDC2(Istanbul_Manual)_	BS	200.00000	1800.00000	-Inf	200.00000	-Inf	-Inf
			.	2000.00000	1484.00000	2.14667	251108.30812
c_u_eMaxDC2(Istanbul_Control)_	BS	1100.0000	900.00000	-Inf	588.57143	-1.22000	249336.97479
			.	2000.00000	1658.57143	.16000	250854.97479
c_u_eMaxDC2(Istanbul_Sensor)_	NU	2000.0000	.	-Inf	1100.00000	-4.08000	247006.97479
			4.08000	2000.00000	2250.00000	+Inf	251698.97479
c_u_eMaxDC2(Ankara_Aliminum)_	BS	.	2000.00000	-Inf	.	-Inf	250678.97479
			.	2000.00000	.	.40000	250678.97479
c_u_eMaxDC2(Ankara_CarbonFiber)_	BS	1132.8571	867.14286	-Inf	100.00000	-.42222	250200.65733
			.	2000.00000	1900.00000	.82133	251609.42812
c_u_eMaxDC2(Ankara_Manual)_	BS	200.00000	1800.00000	-Inf	200.00000	-Inf	-Inf
			.	2000.00000	1003.33333	3.82286	251443.54622
c_u_eMaxDC2(Ankara_Control)_	BS	30.00000	1970.00000	-Inf	30.00000	-Inf	-Inf
			.	2000.00000	833.33333	.54286	250695.26050
c_u_eMaxDC2(Ankara_Sensor)_	NU	2000.0000	.	-Inf	1463.00000	-6.20571	247346.50622
			6.20571	2000.00000	2723.00000	+Inf	255165.70622
c_u_eMaxDC2(Izmir_Aliminum)_	BS	.	2000.00000	-Inf	.	-Inf	250678.97479
			.	2000.00000	.	.56000	250678.97479
c_u_eMaxDC2(Izmir_CarbonFiber)_	NU	2000.0000	.	-Inf	153.33333	-.04706	250592.07283
			.04706	2000.00000	2403.00000	+Inf	250697.93950
c_u_eMaxDC2(Izmir_Manual)_	BS	100.00000	1900.00000	-Inf	100.00000	-Inf	-Inf
			.	2000.00000	455.58824	4.16000	251094.97479
c_u_eMaxDC2(Izmir_Control)_	BS	370.58824	1629.41176	-Inf	15.00000	-3.84533	249253.93950
			.	2000.00000	2091.17647	.05333	250698.73950
c_u_x180_	NU	2000.0000	.	-Inf	268.75000	-6.70353	239073.48950
			6.70353	2000.00000	2377.81250	+Inf	253211.65200

Appendix 14. Python Codes of Revenue- Cost Distribution

```python
#  WE WANNA BONUS...
# For this reason:

# REVENUE/COST DISTRIBUTIONS:

# REVENUES:
# Revenue from Selling components
revenueFromSellingComponents = sum(mdl.vX[i, j, k].value * mdl.pSPC[i, j, k] for i in mdl.I for j in mdl.J for k in mdl.K)
revenueFromSellingComponents = int(revenueFromSellingComponents)
# Revenue from Selling Robotic Kits
revenueFromSellingRoboticKits = sum(mdl.vY[i, k].value * mdl.pSPR[i, k] for i in mdl.I for k in mdl.K)
revenueFromSellingRoboticKits = int(revenueFromSellingRoboticKits)

# Production Costs:
# Production Cost of Components
productionCostofComponents = sum(mdl.vX[i, j, k].value * (mdl.pPCC[i, j, k] * (1.12 if k == 2 else 1.0)) for i in mdl.I for j in mdl.J for k in mdl.K)
productionCostofComponents = int(productionCostofComponents)
# Production Cost of Robotic Kits
productionCostofRoboticKits = sum(mdl.vY[i, k].value * (mdl.pPCR[i, k] * (1.12 if k == 2 else 1.0)) for i in mdl.I for k in mdl.K) - sum(mdl.vY[i, k].value * sum(mdl.pPCC[i, j, k] * mdl.pR[j] * (1.12 if k == 2 else 1.0) for j in mdl.J) for i in mdl.I for k in mdl.K)
productionCostofRoboticKits = int(productionCostofRoboticKits)

# Inventory Holding Costs:
# Inventory Holding Cost of Components
inventoryCostofComponents = 0.08 * sum(mdl.vI[i, j, k].value * mdl.pPCC[i, j, k] for i in mdl.I for j in mdl.J for k in mdl.K)
inventoryCostofComponents = int(inventoryCostofComponents)
# Inventory Holding Cost of Robotic Kits
inventoryCostofRoboticKits = 0.08 * sum(mdl.vIR[i, k].value * mdl.pPCR[i, k] for i in mdl.I for k in mdl.K)
inventoryCostofRoboticKits = int(inventoryCostofRoboticKits)


totalRevenues = revenueFromSellingComponents + revenueFromSellingRoboticKits
totalProductionCost = productionCostofComponents + productionCostofRoboticKits
totalInventoryCost = inventoryCostofComponents + inventoryCostofRoboticKits
totalCosts = totalProductionCost + totalInventoryCost

# Pie Chart of Revenue-Cost Distribution
revCosts = [f"Total Revenues :{totalRevenues}", f"Total Costs: {totalCosts}"]
valuesRC = [totalRevenues, totalCosts]
labelsRC = ['Total Revenues', 'Total Costs']
pembiş = ['pink','red']
plt.figure(figsize=(7, 4))
wedges, texts, autotexts = plt.pie(valuesRC,labels=revCosts, startangle=90,autopct='%1.2f%%',textprops={'fontsize': 12},colors=pembiş)
plt.legend(wedges,labelsRC,title="Revenue-Cost Distribution",loc="lower right",bbox_to_anchor=(1, 0, 0.5, 1),fontsize=12)
plt.title('Revenue - Cost Distribution', fontsize=14)
plt.savefig('chart1.jpg', format='jpg', dpi=300)
plt.show()

# BAR CHART of COSTS
costs = [f"Total Production Cost of Components:{productionCostofComponents}", f"Total Production Cost of Robotic Kits:{productionCostofRoboticKits}", f"Total Inventory Cost of Components: {inventoryCostofComponents}", f"Total Inventory Cost of Robotic Kits: {inventoryCostofRoboticKits}"]
valuesCosts = [productionCostofComponents,productionCostofRoboticKits,inventoryCostofComponents,inventoryCostofRoboticKits]
labelsCosts = ["Total Production Cost of Components", "Total Production Cost of Robotic Kits", "Total Inventory Cost of Components", "Total Inventory Cost of Robotic Kits"]
tonSürTonTon = ['#006400', '#228B22', '#32CD32', '#98FB98'] #özel istek üzerine yaptık ama son 3 tanesi 0 :(

plt.figure(figsize=(8, 5))
plt.bar(labelsCosts, valuesCosts, color=tonSürTonTon)
plt.title('Cost Distribution', fontsize=14)
plt.ylabel('Costs', fontsize=12)
plt.xticks(rotation=45, ha='right', fontsize=10)
plt.tight_layout()
plt.savefig('chart2.jpg', format='jpg', dpi=300)
plt.show()

# BABA NEREDEN GELİYOR BU DEĞİRMENİN SUYU?
# PİE CHART of REVENUES
revenues = [f"Total Revenue from Selling Components: {revenueFromSellingComponents}", f"Total Revenue from Selling Robotic Kits: {revenueFromSellingRoboticKits}"]
valuesRev = [revenueFromSellingComponents, revenueFromSellingRoboticKits]
labelsRev = ["Total Revenue from Selling Components", "Total Revenue from Selling Robotic Kits"]
blues = ["blue", "red"]

plt.figure(figsize=(12, 4))
wedges, texts, autotexts = plt.pie(valuesRev,labels=revenues, startangle=90,autopct='%1.2f%%',textprops={'fontsize': 12},colors=pembiş)
plt.legend(wedges,labelsRev,title="Revenue Distribution",loc="upper left",bbox_to_anchor=(1, 0, 0.5, 1),fontsize=12)
plt.title('Revenue - Cost Distribution', fontsize=14)
plt.savefig('chart3.jpg', format='jpg', dpi=300)
plt.show()
# Component satışından geliyormuş xD
```

Appendix 15. Python Codes of Plant, Component and Period Distributions

```python
# ŞİMDİ ÜRETİMİN PLANT, COMPONENT ve PERİYOTLARA GÖRE DAĞILIMI VAR...

# Production according to the plants
angara = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if i == "Ankara") + sum(mdl.vY[i,k].value for i in mdl.I for k in mdl.K if i == "Ankara")
angara = int(angara)
# ULAN İZzzDANBUL SEN Mİ BÜYÜKSÜN BEN Mİ???
#                                   -ANKARA
izdanbul = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if i == "Istanbul") + sum(mdl.vY[i,k].value for i in mdl.I for k in mdl.K if i == "Istanbul")
izdanbul = int(izdanbul)
ızmir = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if i == "Izmir") + sum(mdl.vY[i,k].value for i in mdl.I for k in mdl.K if i == "Izmir")
ızmir = int(ızmir)
print("Plants : ", angara,izdanbul,ızmir)
# Pie Chart of Plant Distribution
pla = [f"Ankara :{angara}", f"İstanbul : {izdanbul}", f"İzmir : {ızmir}"]
valuesPla = [angara,izdanbul,ızmir]
labelsPla = ["Ankara","İstanbul","İzmir"]

plt.figure(figsize=(7, 4))
wedges, texts, autotexts = plt.pie(valuesPla,labels=pla, startangle=90,autopct='%1.2f%%',textprops={'fontsize': 12})
plt.legend(wedges,labelsPla,title="Plant Distribution",loc="lower right",bbox_to_anchor=(1, 0, 0.5, 1),fontsize=12)
plt.title('Distribution of Product according to the Plants', fontsize=14)
plt.savefig('chart4.jpg', format='jpg', dpi=300)
plt.show()
# İstanbul daha büyükmüş zaaa

# Production according to the components
aluminyum = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if j == "Aliminum")
aluminyum = int(aluminyum)
karbonFiber = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if j == "CarbonFiber")
karbonFiber = int(karbonFiber)
manuelModül = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if j == "Manual")
manuelModül = int(manuelModül)
advContModül = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if j == "Control")
advContModül = int(advContModül)
advSensModül = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if j == "Sensor")
advSensModül = int(advSensModül)
robocob = sum(mdl.vY[i,k].value for i in mdl.I for k in mdl.K)
robocob = int(robocob)
print("Components: ",aluminyum,karbonFiber,manuelModül,advContModül, advSensModül, robocob)

# Pie Chart of Component Distribution
comP = [f"Aluminum :{aluminyum}", f"Carbon Fiber Frame: {karbonFiber}", f"Manuel Modules: {manuelModül}",f"Adavnced Control Modules: {advContModül}", f"Advanced Sensor Modules: {advSensModül}", f"Robocob: {robocob}"]
valuesComp = [aluminyum,karbonFiber,manuelModül,advContModül, advSensModül, robocob]
labelsComp = ["Aliminum", "Carbon Fiber Frames", "Manual Modules", "Advanced Control Modules", "Advanced Sensor Modules", "Robotic Kit"]

plt.figure(figsize=(11, 4))
wedges, texts, autotexts = plt.pie(valuesComp,labels=comP, startangle=90,autopct='%1.2f%%',textprops={'fontsize': 12})
plt.legend(wedges,labelsComp,title="Component Distribution",loc="lower right",bbox_to_anchor=(1.7, 0.5, 0.5, 2),fontsize=12)
plt.title('Distribution of Product according to the Types ', fontsize=14)
plt.savefig('chart5.jpg', format='jpg', dpi=300)
plt.show()

# Production according to the period
ilkay = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if k ==1) + sum(mdl.vY[i,k].value for i in mdl.I for k in mdl.K if k ==1)
ilkay = int(ilkay)
ikinciay = sum(mdl.vX[i,j,k].value for i in mdl.I for j in mdl.J for k in mdl.K if k ==2) + sum(mdl.vY[i,k].value for i in mdl.I for k in mdl.K if k ==2)
ikinciay = int(ikinciay)
print("Period: ",ilkay,ikinciay)

# Pie Chart of Period Distribution
per = [f"First Period :{ilkay}", f"Second Period: {ikinciay}"]
valuesPer = [ilkay,ikinciay]
labelsPer = ['First Period', 'Second Period']

plt.figure(figsize=(6.6, 4))
wedges, texts, autotexts = plt.pie(valuesPer,labels=per, startangle=90,autopct='%1.2f%%',textprops={'fontsize': 12})
plt.legend(wedges,labelsPer,title="Period Distribution",loc="lower right",bbox_to_anchor=(1, 0, 0.5, 1),fontsize=12)
plt.title('Distribution of Product according to Periods', fontsize=14)
plt.savefig('chart6.jpg', format='jpg', dpi=300)
plt.show()
```