Functionality	Input	Output	Queries	File	Program Interface
Mapping tables	1	2	1	5	1
Assigning tables	1	1	1	10	1
Payment System	2	2	1	2	3
Ordering System	1	3	0	1	2

	Complexity				
Description	Total #	Low	Medium	High	Total
Inputs	**5**	**2*3**	**2*4**	**1*6**	**20**
Outputs	**8**	**4*4**	**2*5**	**2*7**	**40**
Queries	**3**	**1*7**	**1*10**	**1*15**	**32**
Files	**18**	**5*7**	**10*10**	**3*15**	**180**
Program Interface	**7**	**1*5**	**5*7**	**1*10**	**50**
Total Unadjusted Function Point (TUFP):					322

The total processing complexity (PC):

Complexity is from 0 to 3: (0=no effect on project complexity; 3=great effect on project complexity)

Tasks	Complexity (0-3)
Data communication	2
Team cohesion	1
Familiarity with technology	3
Data entry	3
Total Processing Complexity (TPC):	9

- **The adjusted processing complexity (APC):**
  APC = 0.65 + (0.01 * TPC) = 0.74

- **Total adjusted function points (TAFP):**
  TAFP = TUFP * APC = 322*0.74 = 238.28

- **Converting Function Points to Lines of Code (LoC):**

Language/Tool	#of LoC/FP
nodeJS	10
Firebase	30
Bootstrap	20
Express	25

- 40% in nodeJS
- 20% in Firebase
- 15% in Bootstrap
- 15% in express

- **Number of lines of code (LoC) = TAFP * # of (LOC/FP) * %**

  For nodeJS= (238.28) * (10) * (.4) = 953LoC

  For Firebase= (238.28) * (30) * (.2) = 1429LoC

  For Bootstrap= (238.28) * (20) * (.15) = 714LoC

  For Express= (238.28) * (15) * (.15) = 536LoC

  Hence, total LoC = 3632 LoC

- **Estimating the effort:**
  Effort=2.4*LoC/1000
  =2.4*3632/1000
  =8.716personMonth

- **Estimating the schedule time:**
  Time = 2.5 * $(effort)^{0.38}$
  =2.5 * $(8.716)^{0.38}$=5.69

- **Estimating the number of persons:**
  Average # of person = effort / time
  = 8.716/5.69 = 1.53