Chapter 1

Project Estimations

In this chapter we are going to estimate the main features of *myTaxiService* project, by using COCOMO II. Reading from the reference manual:

> *The COCOMO II model is part of a suite of Constructive Cost Models. This suite is an effort to update and extend the well-known COCOMO (Constructive Cost Model) software cost estimation model originally published in Software Engineering Economics by Barry Boehm in 1981.*

In the section 1.1 we focus on the project's size in term of lines of code, whereas in the section 1.2 other metrics, such the required time and the costs will be analysed.

1.1 Project Size (Function Points)

Reading from the reference manual:

> *The function point cost estimation approach is based on the amount of functionality in a software project and a set of individual project factors [Behrens 1983; Kunkler 1985; IFPUG 1994]. Function points are useful estimators since they are based on information that is available early in the project life-cycle. A brief summary of function points and their calculation in support of COCOMO II follows.*

The function types are five, described in the table[1].

Function Point	**Description**
External Input (EI)	Count each unique user data or user control input type that enters the external boundary of the software system being measured.
External Output (EO)	Count each unique user data or control output type that leaves the external boundary of the software system being measured.
Internal Logical File (ILF)	Count each major logical group of user data or control information in the software system as a logical internal file type. Include each logical file (e.g., each logical group of data) that is generated, used, or maintained by the software system.
External Interface Files (EIF)	Files passed or shared between software systems should be counted as external interface file types within each system.
External Inquiry (EQ)	Count each unique input-output combination, where input causes and generates an immediate output, as an external inquiry type.

Finally, to perform the analysis we have to present other two tables from the same reference manual of the other one. The first one will be used to classify each function on three level of complexity (high, medium and low).

The second one shows the weights to be used into the estimation formulas[2].

[1]The table is given by the COCOMO II reference manual.

[2]The UFP acronym means Unadjusted Function Points

Table 2. FP Counting Weights

For Internal Logical Files and External Interface Files

	Data Elements		
Record Elements	**1 - 19**	**20 - 50**	**51+**
1	Low	Low	Avg.
2 - 5	Low	Avg.	High
6+	Avg.	High	High

For External Output and External Inquiry

	Data Elements		
File Types	**1 - 5**	**6 - 19**	**20+**
0 or 1	Low	Low	Avg.
2 - 3	Low	Avg.	High
4+	Avg.	High	High

For External Input

	Data Elements		
File Types	**1 - 4**	**5 - 15**	**16+**
0 or 1	Low	Low	Avg.
2 - 3	Low	Avg.	High
3+	Avg.	High	High

Table 3. UFP Complexity Weights

	Complexity-Weight		
Function Type	**Low**	**Average**	**High**
Internal Logical Files	7	10	15
External Interfaces Files	5	7	10
External Inputs	3	4	6
External Outputs	4	5	7
External Inquiries	3	4	6

Up to now, we have presented the Function Points technique. Now, we are going to start our analysis, split by the function type.

1.1.1 Internal Logic Files

The system has to manage Internal Logic Files to store information related to the users (both *normal* and drivers), the *historical* rides, the areas and the driver workshifts[3]
The users have from 12 to 16 fields to be stored (the second number is referred to the drivers case) and only the alerts and the zerotime or future rides have to be stored, thus the complexity is low. The areas and the workshifts can also be considered as low complexity type. In fact they have a few fields and less than six extra records.
The rides have 10 fields, including two positions, the driver and the passenger, all saved in separate entities. They can be considered as an average complexity type, since we have about seven records per ride (in fact in addition to the five presented, the positions requires additional records).

In the table the analysis is summarize:

ILF	Complexity	FP
User	Low	7
Area	Low	7
Workshift	Low	7
Ride	Average	10
Total		31

1.1.2 External Logic Files

The system acquires data from te GPS interface. A GPS position is essentially a tuple of type Position, described in our database. Hence, we have a low complexity type and 5 FP.

[3]See the logic schema at the page 10 of the Design Document to have a detailed description of each part of the database.

1.1.3 External Inputs

The possible interactions between the users and the system, defined in the RASD, are now quickly described in terms of complexity:

- Registration: this operation is performed only by simple user (not a driver) and involves one data type, the one related to the new user. Its complexity is low;
- Login/Logout/Profile Management: these operations are simple due to one entity only is involved, so the complexity is low;
- Workshift Management: this operation requires to interact to 2 entities (driver and work shift) and can involved more than 20 elements to perform the validity checks. Hence the complexity is high;
- Read the Alerts: this operation has an average complexity since it may have more than 20 elements to be managed;
- Start Waiting Time/End of a ride: these operation requires to interact with three types of files (Position, Area and Driver Waiting) with one element per type, thus the complexity is low;
- Check the Reservations: this is a group of three related operations (shows the alerts and gives the possibilities to modify or to cancel a ride) that involve one type and potentially more than 20 elements, so the complexity is average;
- Book a zerotime/future ride: these operations involve many types of data and several elements, so the complexity is high.

In the following table we have summarized the results:

EI	Complexity	FP
Registration	Low	3
Login/Logout/Profile Management	Low	3x3
Workshift Management	High	6
Read the Alerts	Average	4
Start Waiting Time / End of a Ride	Low	2x3
Check the Reservations	Average	3x4
Book a ride (zerotime or future)	High	2x6
Total		52

1.1.4 External Inquiries

1.1.5 External Outputs

1.2 Effort Estimation (COCOMO II)