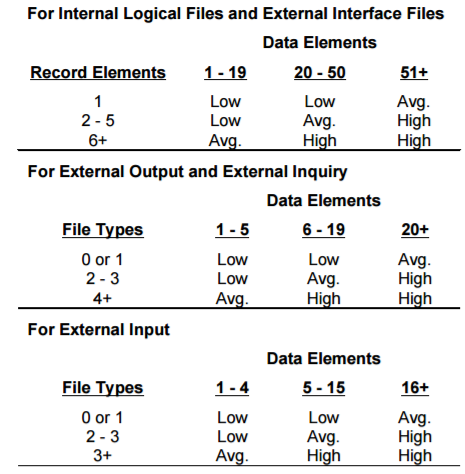
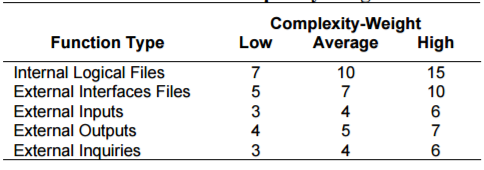
1. **FP estimation**
   1. **Complexity levels and function points**

The table below (from COCOMO II documentation) shows how to evaluate weights of Function Points, based on

the number of data elements:



According to these criteria, the next table (from COCOMO II documentation) shows the numeric weights for the functions:



* 1. **Internal logic files**

An Internal Logic File (ILF) is a homogeneous set of data used and managed by the application.

In our system, we can consider the database tables where we store persistent information as ILFs; the tables are the following:

* User
* Reservation
* Car
* Supervisor
* Safe area
* Safe park slot

|  |  |  |
| --- | --- | --- |
| **ILF** | **Complexity** | **FP** |
| User | AVG | 10 |
| Reservation | AVG | 10 |
| Car | AVG | 10 |
| Supervisor | LOW | 7 |
| Safe Area | AVG | 10 |
| Safe park slot | AVG | 10 |

* 1. **External interface files**

An External Interface File (EIF) is a homogeneous set of data used by the application but generated and maintained by other applications.

The only external application our system interact with is:

* Google Maps

|  |  |  |
| --- | --- | --- |
| **ELF** | **Complexity** | **FP** |
| Maps | HIGH | 10 |

* 1. **External input**

An External Input (EI) is an elementary operation to elaborate data coming from the external environment.

In our system, we can consider as EIs the following:

* User registration
* User login
* Request for car’s research
* Request for car’s reservation
* Request for car’s unlock
* Car’s data (from sensors of physical cars)

|  |  |  |
| --- | --- | --- |
| **EI** | **Complexity** | **FP** |
| User registration | AVG | 4 |
| User login | LOW | 3 |
| Request for car’s research | AVG | 4 |
| Request for car’s reservation | LOW | 3 |
| Request for car’s unlock | LOW | 3 |
| Car’s data | HIGH | 6 |

* 1. **External output**

An External Output (EO) is an elementary operation that generates data for the external environment and it usually includes the elaboration of data from logic files.

In our system, the EOs are:

* Confirmation email for registration
* Results of car research
* Lock/unlock commands to physical car
* Payment requests
* Confirmation email for payment

|  |  |  |
| --- | --- | --- |
| **EO** | **Complexity** | **FP** |
| Confirmation email reg. | LOW | 4 |
| Results of car research | AVG | 5 |
| Lock/unlock commands | LOW | 4 |
| Payment requests | AVG | 5 |
| Confirmation email payment | LOW | 4 |

* 1. **External Inquiry**

An External Inquiry (EQ) is an elementary operation that involves input and output, without significant elaboration of data from logic files.

In our system the EQs are:

* Notifications about reservation (to user and to physical car)
* Data requests from supervisors

|  |  |  |
| --- | --- | --- |
| **EQ** | **Complexity** | **FP** |
| Notifications | AVG | 4 |
| Data request | LOW | 3 |

* 1. **Final estimation**

We can now calculate the total amount of FP of the system:

|  |  |
| --- | --- |
| **Function type** | **FP** |
| Internal logic files | 57 |
| External interface files | 10 |
| External Input | 23 |
| External output | 22 |
| External inquiries | 7 |
| **TOTAL** | **119** |

According to the COCOMO II documentation, the multiplicator to convert the Function Points to Source Lines of Code for Java is 53, so we have:

**SLOC = 53 x 119 = 6307**

Bibliography

http://csse.usc.edu/csse/research/COCOMOII/cocomo2000.0/CII\_modelman2000.0.pdf