**5 Effort Estimation**

First we need to calculate the Use Case Points (UCP)

UCP = UUCP \* TCP \* ECF (5.1)

Where Unadjusted Use Case Points (UUCPs) are computed as a sum of these two components:

1. The Unadjusted Actor Weight (UAW), based on the combined complexity of all the actors in all the use cases.

2. The Unadjusted use Case Weight (UUCW), based on the total number of activities (or steps) contained in all the use case scenarios.

**Unadjusted Actor Weight (UAW) and Unadjusted Use Case Weight (UUCW)**

|  |  |  |
| --- | --- | --- |
| **Actor** | **Complexity** | **Weight** |
| **Users** | **Complex** | **3** |
| **Experts** | **Complex** | **3** |

UAW = 3 + 3 =5 (5.2)

Now we reference the Use Case Traceability Matrix table to calculate the UUCW.

UUCW = x + x + x + x + x + x + x + x = y(5.3)

There the UUCP is: UUCP = 6 + y = (5.4)

Technical Complexity Factor (TCF)-Nonfunctional Requirements

The TCF is one of the factors applied to the estimated size of the software in order to account for technical considerations of the system. It is determined by assigning a score between 0 (factor is irrelevant) and 5 (factor is essential) to each of the 13 technical factors listed in the table below. This score is then multiplied by the defined weighted value for each factor. The total of all calculated values is the technical factor (TF). The TF is then used to compute the TCF with the following formula:

TCF = 0.6 + (TF/100)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Technical Factor** | **Description** | **Weight** | **Perceived Complexity** | **Calculated Factor** |
| **T1** | Friendly interface that the user understands | **2** | **1** | **2** |
| **T2** | Good Performance | **1** | **2** | **2** |
| **T3** | Security is a minor concern | **1** | **2** | **2** |
| **T4** | Ease of use is very important | **3** | **3** | **9** |
| **T5** | User reserving a room and selecting an expert shouldn’t be complex | **2** | **3** | **6** |
|  | Technical Factor Total (TFT) |  |  | **21** |

And TCF = C1 + C2 x TFT, and C1 = 0.6, C2 = 0.01, so

TCF = 0.6 + (0.01 ∗ 21) = 0.126

**Environment Complexity Factor (ECF)**

The ECF is another factor applied to the estimated size of the software in order to account for environmental considerations of the system. It is determined by assigning a score between 0 (no experience) and 5 (expert) to each of the 8 environmental factors listed in the table below. This score is then multiplied by the defined weighted value for each factor. The total of all calculated values is the environment factor (EF). The EF is then used to compute the ECF with the following formula:

ECF = 1.4 + (-0.03 x EF)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Environmental Factor** | **Description** | **Weight** | **Assigned Value** | **Weight x Assigned Value** |
| **E1** | Mostly beginners at UML based development | 2 | 3 | 6 |
| **E2** | Quite knowledgable about the Object-Oriented approach | 2 | 3 | 6 |
| **E3** | Progamming language proficiency | 2 | 3 | 6 |
| **E4** | Motivation of the team | 1 | 2 | 2 |
| **E5** | Lead analyst capability | 0.5 | 3 | 1.5 |
|  | Environmental Factor Total (EFT) |  |  | 21.5 |
|  |  |  |  |  |

ECF = C1 + C2 ∗ EFT

Where C1 = 1.4, C2 = 0.03. Therefore we calculate the ECF. ECF = 1.4 + (−0.03 ∗ 21) = 0.77 (5.7)

So we calculate the final UCP: UCP = 0.126 \* 0.77 \* y = z (5.8)

If we assume that productivity factor is 24 hours per user case point. The effort estimation would be 24 \* z =.