# IN THE NAME OF GOD

## USE CASE POINT

**Producers :**

*Sayyed Hossein Hosseini*

**Supervisor :**

*Dr Mohammad Reza Sharbaf*

**Teacher assistant :**

*Reza Pour Mohammadi*

*Faculty of Computer Engineering, University of Isfahan*

*Autumn 2023*

# TABLE OF CONTENTS

## Definition

**Use case points** (**UCP** or **UCPs**) is a software estimation technique used to forecast the software size for software development projects. UCP is used when the Unified Modeling Language (UML) and Rational Unified Process (RUP) methodologies are being used for the software design and development. The concept of UCP is based on the requirements for the system being written using use cases, which is part of the UML set of modeling techniques. The software size (UCP) is calculated based on elements of the system use cases with factoring to account for technical and environmental considerations. The UCP for a project can then be used to calculate the estimated effort for a project.

## Method

The method for determining the size estimate to develop a system is based on a calculation with the following elements:

- Unadjusted Use Case Weight (UUCW) – the point size of the software that accounts for the number and complexity of use cases.

- Unadjusted Actor Weight (UAW) – the point size of the software that accounts for the number and complexity of actors.

- Technical Complexity Factor (TCF) – factor that is used to adjust the size based on technical considerations.

- Environmental Complexity Factor (ECF) – factor that is used to adjust the size based on environmental considerations.

Once the previous four elements have been calculated, the final size estimate can be calculated. This final number is known as the use case points or UCP for a software development project.

The following sections walk through the various calculations to determine the UCP for a project.

## ❖ Unadjusted Use Case Weight (UUCW)

  The UUCW is one of the factors that contribute to the size of the software being developed. It is calculated based on the number and complexity of the use cases for the system. To find the UUCW for a system, each of the use cases must be identified and classified as Simple, Average or Complex based on the number of transactions the use case contains. Each classification has a predefined weight assigned. Once all use cases have been classified as simple, average or complex, the total weight (UUCW) is determined by summing the corresponding weights for each use case. The following chart shows the different classifications of use cases based on the number of transactions and the weight value assigned for each use case within the classification.

| Use Case Classification | No. of Transactions | Weight |
|---|---|---|
| Simple | 1 to 3 transactions | 5 |
| Average | 4 to 7 transactions | 10 |
| Complex | 8 or more transactions | 15 |

UUCW = (Total No. of Simple Use Cases x 5) + (Total No. Average Use Cases x 10) + (Total No. Complex Use Cases x 15)

❖ **Unadjusted Actor Weight (UAW)**

The UAW is another factor that contributes to the size of the software being developed. It is calculated based on the number and complexity of the actors for the system. Similar to finding the UUCW, each of the actors must be identified and classified as Simple, Average or Complex based on the type of actor. Each classification also has a predefined weight assigned. The UAW is the total of the weights for each of the actors. The following chart shows the different classifications of actors and the weight value assigned.

| Actor Classification | Type of Actor | Weight |
|---|---|---|
| Simple | External system that must interact with the system using a well-defined API | 1 |
| Average | External system that must interact with the system using standard communication protocols (e.g. TCP/IP, FTP, HTTP, database) | 2 |
| Complex | Human actor using a GUI application interface | 3 |

UAW = (Total No. of Simple actors x 1) + (Total No. Average actors x 2) + (Total No. Complex actors x 3)

❖ **Technical Complexity Factor (TCF)**

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:

$$TCF = 0.6 + (TF/100)$$

| Factor | Description | Weight |
|--------|-------------|--------|
| T1 | Distributed system | 2.0 |
| T2 | Response time/performance objectives | 1.0 |
| T3 | End-user efficiency | 1.0 |
| T4 | Internal processing complexity | 1.0 |
| T5 | Code reusability | 1.0 |
| T6 | Easy to install | 0.5 |
| T7 | Easy to use | 0.5 |
| T8 | Portability to other platforms | 2.0 |
| T9 | System maintenance | 1.0 |
| T10 | Concurrent/parallel processing | 1.0 |
| T11 | Security features | 1.0 |
| T12 | Access for third parties | 1.0 |
| T13 | End user training | 1.0 |

❖ **Environmental 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:

$$ECF = 1.4 + (-0.03 \times EF)$$

| Factor | Description | Weight |
|--------|-------------|--------|
| E1 | Familiarity with development process used | 1.5 |
| E2 | Application experience | 0.5 |
| E3 | Object-oriented experience of team | 1.0 |
| E4 | Lead analyst capability | 0.5 |
| E5 | Motivation of the team | 1.0 |
| E6 | Stability of requirements | 2.0 |
| E7 | Part-time staff | -1.0 |
| E8 | Difficult programming language | -1.0 |

❖ **Use Case Points (UCP)**

Finally the UCP can be calculated once the unadjusted project size (UUCW and UAW), technical factor (TCF) and environmental factor (ECF) have been determined. The UCP is calculated based on the following formula:

$$UCP = (UUCW + UAW) \times TCF \times ECF$$

# Example

  To illustrate the process of calculating the UCP, an Online Shopping System will be used. The diagram below depicts the Use Case Diagram for the system to be developed.
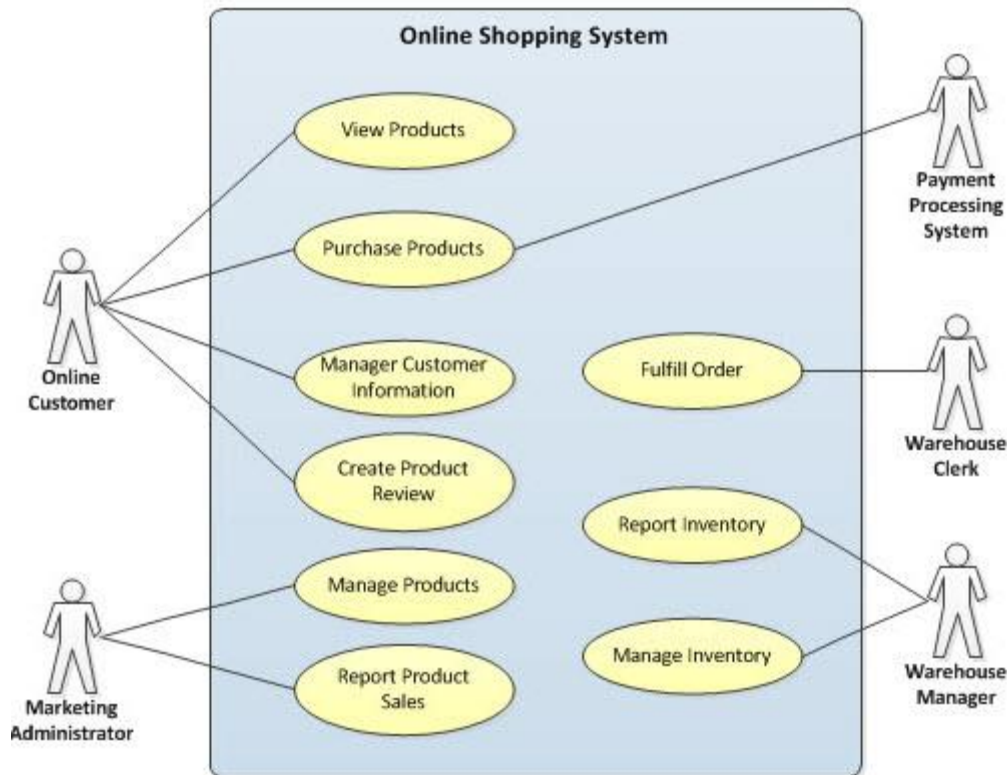


❖ **Unadjusted Use Case Weight (UUCW)**

To calculate the UUCW, the use cases must be defined and the number of transactions for each use case identified. The Online Shopping System use case diagram is depicting that nine use cases exist for the system. Assuming 2 of these use cases are simple, 3 are average and 4 are complex, the calculation for UUCW is as follows:

UUCW = (Total No. of Simple Use Cases x 5) + (Total No. Average Use Cases x 10) + (Total No. Complex Use Cases x 15)

For the Online Shopping System, the UUCW = (2 x 5) + (3 x 10) + (4 x 15) = 100

UUCW = 100

### ❖ Unadjusted Actor Weight (UAW)

To calculate the UAW, the actors must be identified. The Online Shopping System use case diagram is depicting five actors; One simple for the Payment Processing System and four complex for each of the human users actors (i.e. Online Customer, Marketing Administrator, Warehouse Clerk, Warehouse Manager.) The calculation for UAW is as follows:

UAW = (Total No. of Simple Actors x 1) + (Total No. Average Actors x 2) + (Total No. Complex Actors x 3)

For the Online Shopping System, UAW = (1 x 1) + (0 x 2) + (4 x 3) = 13

UAW = 13

### ❖ Technical Complexity Factor (TCF)[edit]

To calculate the TCF, each of the technical factors is assigned a value based on how essential the technical aspect is to the system being developed. The diagram below shows the assigned values for the Online Shopping System. The values are multiplied by the weighted values and the total TF is determined.

| Factor | Description | Weight | Assigned Value | Weight x Assigned Value |
|--------|-------------|--------|----------------|-------------------------|
| T1 | Distributed system | 2.0 | 5 | 10 |
| T2 | Response time/performance objectives | 1.0 | 5 | 5 |
| T3 | End-user efficiency | 1.0 | 3 | 3 |
| T4 | Internal processing complexity | 1.0 | 2 | 2 |
| T5 | Code reusability | 1.0 | 3 | 3 |
| T6 | Easy to install | 0.5 | 1 | 0.5 |

| T7 | Easy to use | 0.5 | 5 | 2.5 |
|-----|-----|-----|-----|-----|
| T8 | Portability to other platforms | 2.0 | 2 | 4 |
| T9 | System maintenance | 1.0 | 2 | 2 |
| T10 | Concurrent/parallel processing | 1.0 | 3 | 3 |
| T11 | Security features | 1.0 | 5 | 5 |
| T12 | Access for third parties | 1.0 | 1 | 1 |
| T13 | End user training | 1.0 | 1 | 1 |
| **Total (TF):** | | | | **42** |

Next, the TCF is calculated:

$$TCF = 0.6 + (TF/100)$$

For the Online Shopping System, $TCF = 0.6 + (42/100) = 1.02$

$$TCF = 1.02$$

❖ **Environmental Complexity Factor (ECF)**

To calculate the ECF, each of the environmental factors is assigned a value based on the team experience level. The diagram below shows the assigned values for the Online Shopping System. The values are multiplied by the weighted values and the total EF is determined.

| Factor | Description | Weight | Assigned Value | Weight x Assigned Value |
|--------|-------------|--------|----------------|-------------------------|
| E1 | Familiarity with development process used | 1.5 | 3 | 4.5 |
| E2 | Application experience | 0.5 | 3 | 1.5 |
| E3 | Object-oriented experience of team | 1.0 | 2 | 2 |
| E4 | Lead analyst capability | 0.5 | 5 | 2.5 |
| E5 | Motivation of the team | 1.0 | 2 | 2 |
| E6 | Stability of requirements | 2.0 | 1 | 2 |
| E7 | Part-time staff | -1.0 | 0 | 0 |
| E8 | Difficult programming language | -1.0 | 4 | -4 |
| **Total (EF):** | | | | **10.5** |

Next, the ECF is calculated:

ECF = 1.4 + (-0.03 x EF)

For the Online Shopping System, ECF = 1.4 + (-0.03 * 10.5) = 1.085

ECF = 1.085

❖ **Use Case Points (UCP)**

Once the Unadjusted Use Case Weight (UUCW), Unadjusted Actor Weight (UAW), Technical Complexity Factor (TCF) and Environmental Complexity Factor (ECF) has been determined, the Use Case Points (UCP) can be calculated with the following formula:

UCP = (UUCW + UAW) x TCF x ECF

For the Online Shopping System, UCP = (100 + 13) x 1.02 x 1.085 = 125.06

UCP = 125.06

For the Online Shopping System, the total estimated size to develop the software is 125.06 Use Case Points.

Now that the size of the project is known, the total effort for the project can be estimated. For the Online Shopping System example, 28 man hours per use case point will be used.

Estimated Effort = UCP x Hours/UCP

For the Online Shopping System, Estimated Effort = 125.06 x 28

Estimated Effort = 3501 Hours

## Educational Resources

❖ **Video Link**

**Educational Video**