

<https://i.stack.imgur.com/643hJ.png>

The description of the relation is the following:

* A single customer can give away one or more projects
* A single project can be given by ONE SINGLE CUSTOMER
* Each project can be finished by one or more developers
* Each developer can work on multiple projects (regardless of the customer by which the project was given)

The problem is in the ER-Diagram itself: it does not exactly represent the description above. The problem lies in the constraint that a single project can be given by one single customer. That's why it would make more sense to model that with **two separate binary relationships** instead using a ternary one.

That being said, the relationship between *Customer* and *Project* should be a 1:n relationship, while the relationship between *Project* and *Developer* should be a m:n relationship. Mapping those relationships gives us the following:

* Customer(CustomerID) *with Primary Key=CustomerID*
* Project (ProjectID, CustomerID) *with Primary Key=ProjectID and Foreign Key=CustomerID referencing the Customer (Customer\_ID)*
* Developer(DeveloperID) with PK=DeveloperID
* ProjectDevelopment (ProjectID, DeveloperID) with PK={ProjectID, DeveloperID)