LOGICAL MODEL

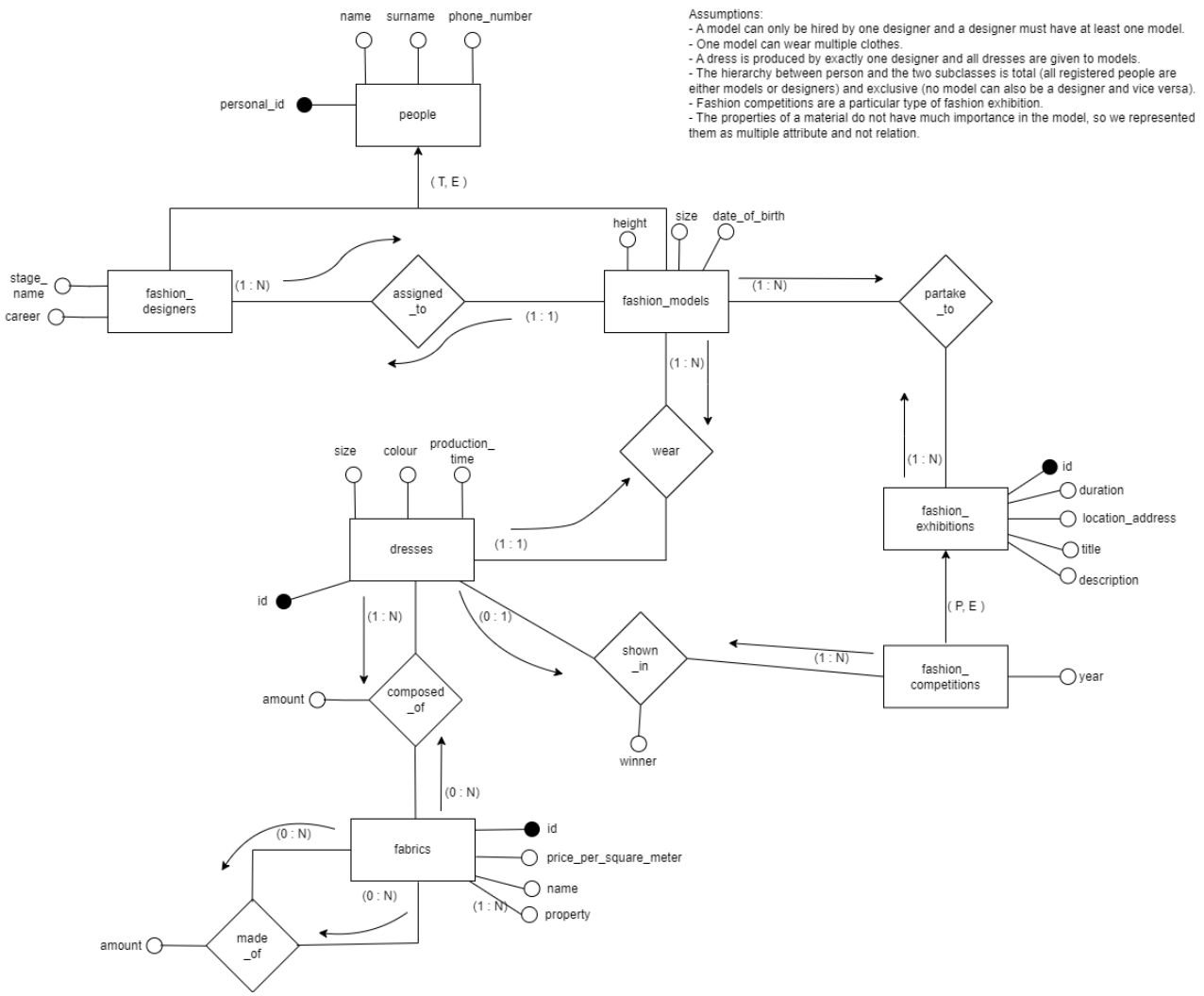

Figure 1: ER model.

After having defined the ER model (Figure 1 above), we now proceed with the definition of the logical model, in which we will define the entities, the attributes, the primary and foreign keys, and finally the relationships between entities.

ENTITIES:

people:

This entity represents the general concept of persona and has two subclasses.

Since the hierarchy is (Total, Exclusive), the parent entity (i.e. people), is not translated into a table.

fashion_designers:

This entity contains the information of fashion designers.

Inherits the following attributes from the people class: personal_id, name, surname, phone_number. Then add the attributes: stage_name, career.

The primary key is: personal_id.

fashion_models:

This entity contains the information of fashion models.

Inherits the following attributes from the people class: personal_id, name, surname, phone_number. Then add the attributes: height, size, date_of_birth, fashion_designers_personal_id.

The primary key is: personal_id.

Foreign key:

- fashion_designers_personal_id: reference to the personal id of the designer to whom this model is assigned.

dresses:

This entity contains information about the dresses.

A dress has the following attributes: id, size, colour, production_time, fashion_models_personal_id.

The primary key is: id.

Foreign key:

- fashion_models_personal_id: reference to the personal id of the model who has worn such dress.

fabrics:

This entity contains information about the fabrics that have been used to make dresses or other fabrics.

A fabrics has the following attributes: id, price_per_square_meter, name.
The multiple attribute "property" is translated into a new table (see below)

The primary key is: id.

fashion_exhibitions:

This entity contains information about fashion exhibitions, and is also the parent entity of fashion_competitions.

Since the hierarchy is (Partial, Exclusive) both the parent entity and the daughter entity are translated.

It has the following attributes: id, duration, location_address, title, description.

The primary key is: id.

fashion_competitions:

This entity contains information about fashion competitions.

Inherits the following attributes from the fashion_exhibitions class: id, duration, location_address, title, description. Then add the attribute: year.

The primary key is: id.

RELATIONSHIP:

assigned_to:

This relationship allows you to assign a model to a designer.

Since the cardinality on the model side is (1:1) (a model is assigned to exactly one designer), this relationship is not translated and a foreign key will be added inside fashion_models, as shown before.

wear:

This relationship connects a dress to the model who wore it.

Since the cardinality on the dress side is (1:1) (a dress is worn by exactly one model), this relationship is not translated and a foreign key will be added inside dresses, as shown before.

partake_to:

This relationship connects fashion models and fashion exhibitions.

Since the cardinality on both side is (1:N), this relationship is translated and it contains two foreign keys.

Primary key: the couple fashion_models_personal_id – fashion_exhibitions_id

Foreign keys:

- fashion_exhibitions_id: reference to the id of the fashion exhibition.
- fashion_models_personal_id: reference to the personal id of the model.

shown_in:

This relationship connects the dresses and the fashion competitions.

Since the cardinality is (1:N) on the fashion competitions side and (0:1) on the dress side with low load, this relationship is translated and it contains two foreign keys and the attribute winner.

Primary key: the couple dresses_id – fashion_competitions_id

Foreign keys:

- dresses_id: reference to the id of the dress shown in the competition.
- fashion_competitions_id: reference to the id of the fashion competition.

composed_of:

This relationship connects dresses and the respective fabrics.

Since the cardinality is (1:N) on the dresses side and (0:N) on the fabrics, this relationship is translated and it contains two foreign keys and the attribute amount.

Primary key: the couple dresses_id – fabrics_id

Foreign keys:

- dresses_id: reference to the id of the dress.
- fabrics_id: reference to the id of the fabric.

made_of:

This relationship connects fabrics with themselves, because a fabric can be make of another fabric.

Since the cardinality is (0:N) on both sides, this relationship is translated and it contains two foreign keys and the attribute amount.

Primary key: the couple fabrics_id1 – fabrics_id2

Foreign keys:

- fabrics_id1: reference to fabric id made from another fabric.
- fabrics_id2: reference to the fabric id used to make another fabric.

properties_of_fabrics:

This relationship connects a fabric to its properties.

Primary key: the couple fabrics_id – properties_name

Foreign key:

- fabrics_id: reference to the id of the fabric.

CONCLUSION:

fashion_designers (personal_id, name, surname, phone_number, stage_name, career)

fashion_models (personal_id, name, surname, phone_number, height, size, date_of_birth, fashion_designers_personal_id)

dresses (id, size, colour, production_time, fashion_models_personal_id)

fabrics (id, price_per_square_meter, name)

fashion_exhibitions (id, duration, location_address, title, description)

fashion_competitions (id, duration, location_address, title, description, year)

partake_to (fashion_exhibitions_id, fashion_models_personal_id)

shown_in (fashion_competitions_id, dresses_id, winner)

composed_of (dresses_id, fabrics_id, amount)

made_of (fabrics_id1, fabrics_id2, amount)

properties_of_fabrics (fabrics_id, properties_name)

Legend:

Underline attribute(s) → Primary key

Green attribute → Foreign key