Changes To ER model:

1. In the subclass Wine/Beer, Wine name/Beer name is redundant to the attribute Drink name given in the superclass Drinks, so they are deleted.
2. In the enitity Customer's Guardian, Customer Id is obtained through the relationship Guards, so the attribute Customer ID is deleted and Guardian name attribute is included as partial key (part of the primary key of Customer's Guardian).
3. In the entity Bar Manager, Favourite Drink name attribute is deleted as it is obtained through the relationship LovestoSell and Name attribute is included as the partial key.
4. In the entity Customer, Favourite Bar ID attribute is removed as LovestoGoto relationship handles that.

Relational model construction:

A relational model is constructed from the ER diagram using the following steps:

Step 1: Mapping strong entity types:

We include all the simple(non multivalued attributes and only components of composite attributes not the composite themselves) attributes of the strong entity in the relation and underline the primary key attribute(s). We map Customer, Bar and Drinks in this manner

Step 2: Mapping weak entity types:

We include all the simple attributes of the weak entity in the relation and also include the primary key attributes of the strong entity that it identifies with as the foreign key. This foreign key together with any pre-existing partial key, together form the new primary key. Bar Manager and Customer's Guardian are mapped this way.(This basically maps the identifying relationships of the weak entities as well)

Step 3: Mapping subclasses:

The distinguishing attributes of the subclasses is included in the relationship made from the superclass. We add attributes Wine brand and Beer brand to Drinks relation to signify the subclass Wine or Beer and also give value to that attribute

Step 4: Mapping multivalued attributes:

The multivalued attributes are taken in a separate relation with the primary key attributes of the entity they belonged to which serves as the foreign key to the relation the entity is mapped to. This foreign key, together with the multivalued attribute itself forms the primary key of the relation.Manager Contact includes the primary key of Bar manager and multi valued attribute contact no. and references Bar manager, Guardian Contact and Bar Contact relations do the exact same thing with Customer's Guardian and Bar relation respectively.

Step 5: Mapping N:1 binary relationship types:

We Include the primary key of the relation on the '1' side of the relationship as the the foreign key for the other relation. We add foreign key Drink Name to Bar Manager relation to reference the relation Drinks, to map the relationship type Loves to Sell.

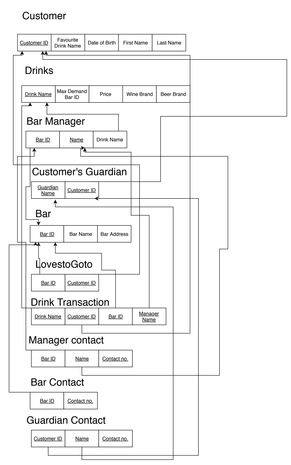
Step 6: Mapping N:M binary relationship types:

We create a new relation which include the primary key of both participating relations as the foreign keys and these foreign keys together form the primary key. LovestoGoto relation references relations Bar and Customer and maps the relationship type of the same name

Step 7: Mapping N-ary relationship types:

Include the primary key attributes of all participating relations as the foreign keys of a new relation which together serve as the primary key for the new relation. Drink transaction references relations Drinks, Customers, Bar Manager and Bar.

Relational data model:



1NF conversion:

For the given relational model, multivalued attributes and composite attributes are handled by the default relational model itself, hence its already 1NF.

2NF conversion:

The given relational model is also 2NF as for every relation with more than 1 primary key attribute, no non-primary key set of attributes is functionally dependent on any subset of the primary key.
eg. Both Bar ID and name is needed for the Drink name in the Bar Manager relation, neither one alone can decide the value of any attribute or set of attributes as 2 distinct managers can have the same name.
Same holds for Customer's Guardian relation
The same is true for all relations in the relational model.

3 NF conversion

The given model is also already 3NF since there is no transitive dependence between any non-prime attribute and the primary key

Example:

Customer

1	Tuborg Strong	12/11/2000	barry	allen	No field
2	Tuborg Green	14/11/2000	bruce	wayne	No field
No field	No field	No field	No field	No field	No field

Drinks

Tuborg Strong	1	1000 rs	NULL	Tuborg	No field
Tuborg Green	2	2000 rs	NULL	Tuborg	No field
No field	No field	No field	No field	No field	No field

Bar Manager

1	Rohan	Tuborg Strong
2	Pranav	Tuborg Green
No field	No field	No field

Customer's Guardian

Rambo	1	No field
John	2	No field
No field	No field	No field

Bar

1	Turbo bar	gachibowli,hyderabad
2	Speed Bar	kondapur,hyderabad
No field	No field	No field

LovestoGoto

1	1	No field
2	2	No field
No field	No field	No field

Drinks transaction

Turbo Strong	1	1	Rohan	No field	No field
Turbo Green	2	2	Pranav	No field	No field
No field	No field	No field	No field	No field	No field

Manager Contact

1	Rohan	1234567890
1	Rohan	0987654321
2	Pranav	1234567098

Bar Contact

1	1212121212	No field
2	1313131313	No field
No field	No field	No field

Guardian Contact

1	Rambo	1234509876
1	Rambo	0987651234
2	John	1209384756

NOTE:

All cells labelled No field in the example means that cell is not supposed to exist in the table, that is that the cell should be ignored, while NULL means some given attribute has no value for that particular tuppple