Domain Model

- **Missing concept of Road/City/Monastery/Field (all of them)**

Added a class named Construction which represents Road/City/Monastery/Village/Road Intersection

- **Does not include distinction between segments (single features on a tile) and aggregate features, as discussed in the writeup.**

One tile contains five objects of Component that describe the five segments (Left/Right/Up/Down /Center) on the tile.

- **Missing concept of number of meeples for player**

Added an attribute called 'Number of Meeples' in Player class.

- **Missing coat of arms/shield for cities**

Added an attribute called "has Shield" in Tile class to represent if there is a shield or not.

- **Missing Tile - Segment (Tile - Feature acceptable, but still not accurate)**

Added Component class.

- **Feature - Segment**

Composition relations between Tile and Component.

- **The modeling of classes vs. attribute in your domain model is unclear**

Add attributes in some classes.

- **Your domain model includes interfaces, abstract classes, or type specifiers on attributes.**

Removed.

- **Names in your domain model do not align with names of real-world concepts.**

Changed.

System Sequence Diagram

- **Missing starting a new game, ideally with the number of players as a parameter**

Add an interaction called Start Game with the number of players.

- **Missing placing a meeple, including relevant parameters: the segment or other designation for meeple position**

Place meeple function has a parameter Direction to describe where the meeple is placed.

Behavioral Contract

- Modelling Accuracy & Level of Abstraction

Modified preconditions and postconditions. Delete the second operation contract.

Interaction Diagram 1

- **The diagram does not demonstrate the steps necessary to confirm a tile placement is valid.**

Added Coordinate and Component classes and some interactions between Main and Board, Board and itself, Board and Coordinate, Board and Tile, Tile and Component.

- **It is not clear how the move validation interaction is initiated by the user.**

User passes the x-coordinate and y-coordinate to Main class through button's ActionListener.

- **Design Aspects & Level of Abstraction**

Have a class called CarcassonneImp to interact with GUI. Board class deals with the placement check. Coordinate class has an attribute containing the neighbor coordinates of the object and can be called by board for placement check. Tile is composed of five Component objects which contain the Road/City/ Monastery/Village/Intersection.

Interaction Diagram 2

The process of users initiating tile placement is described in the first interaction diagram. The Board takes charge of checking if the neighbors have a monastery and if the monastery is completed by examining the neighbors' neighbor coordinates. Once the checking is finished, the scoreboard gets updated and GUI is updated as well.

Object Model:

Corrected the notation issue and redrew the diagram based on the domain model. Added functions in Carcassonne Interface to show the players' order and the number of meeples owned by players. Used an enumeration class Construction to represents Road/City/Monastery/Village/Intersection in a tile and used an enumeration class Direction class to represent the directions of segments in a tile and also the directions of neighbors. The Board class handles the scoring of features.