a – How this design fulfills the requirements of the job?

The implemented database design created fulfills the job requirements put forth. The database is capable of storing varied combat sports data and creating non-trivial statistics.

The design of the database is capable of storing combat sports related data. This is facilitated due to the relational nature of entities such as fights and ruleset. Through this relation, what type of sport the fight is taking place in is specified. Also, the entities Commissions, Promoters, Cards, FightLocations allow for storage of non-fighter information. These entities allow for more in-depth sources of information.

The task of creating the statistics and insights in two ways. First from the data held in the database and its relations to other entities. Second by compounding on available statics to draw a greater statics and insights. The relational nature of a SQL database allows for the discovery of insights into combat sports. Insights such as what referee is the most likely to stop a fight. In the previous example Panels, FightCombatants, Fights, and Refs could be joined. Then queried for the ref with the most referee stoppages. An example of the second method of using existing stats would be listing all fighters by reach for a given weight class. This is done joining Fighters and PhysicalStats. Then sorting by reach where normalWeight is equal to the weight class of choice.

The database flexibility, relational structure, and entities are key to the fulfillment of the job requirements. As these factors allow for the storage and creation of statistics for any combat sport.

b – What issues you faced in creating your design?

The issues that I faced in creating my design were normalization, relation, and cardinality.

Normalization to the third normal form provided challenges in how to best design the entities. Also about whether I was in the third normal form. The latter issue happened at the beginning of the project when the notion of normal forms was not clear. This lead to many instances of redesigning the database and the creation of more tables.

The relational nature of SQL lead to some issues in the database design. A common occurrence when normalizing was to create the relation wrong way. This lead to a design that did not function in a correct manner and did not have useful relations.

The issues with cardinality revolved around many to many, and one to many relationships. The issue with many to many relationships was in the fighter to fights relationship. Not knowing how to handle this relationship resulted in a significant redesign to the database. The solution was a FightCombantans entity that represents the occurrence of a fight. From here it was possible to relate aspects of a fight that are dependant on the fight. One to many relationships provided the challenge at the beginning of the project. Where to implement the relationship as it was not known. Once implemented there were some issues with the 1 to many relationships ending up the wrong way.

c – What areas would you improve if you had more time?

Areas which I would improve include adding various triggers, greater referential integrity and more entities. These areas would have provided the greatest improvement to the project. Yet they had to be sacrificed due to time constraints.

The first aspect not implemented was triggers as knowledge of they came too late put them in to use. One trigger that would have been useful would be when a fight decision is inserted. The trigger would then update the fighters record table. This would provide the advantage of not having to write a complex query to retrieve the most used information. Secondly, there were few other instances where I thought off or could find a use for a trigger in the database.

I would seek to improve the current referential integrity if I had more time. I am not fully satisfied with the amount of inbuilt error checking. Despite having referential integrity I felt as it was still possible for inconsistent data to be entered. All while not having an issue with inconsistent data.

Finally, I would seek to add functionality through more entries to the database. I would have liked there to have been more entities related to commissions, and promoters. I had an initial goal of providing a more general implementation. An implementation that could be used to run a promotion or athletic commission. Yet due to time and not wanting to allow too much scope creep I had to give up implementing the more entities.