1. 1. *Is the service RESTful? If not, why not? If so, what key features make it RESTful.* Yes, our service is RESTful. We have a client/server structure. It will apply JSON and XML. We also might have to do something like the OAuth system used in Twitter’s API if we have unique profiles/logins for each user so that they cannot create rides using someone else’s data.
   2. *What endpoints implement actions that are* idempotent? nullipotent? An endpoint using POST, DELETE, or GET would all be idempotent since they perform the same action each time and receive the same side effects every time they are used (except this is not true if DELETE is used on the same item twice in a row since it will have an error). GET would be nullipotent since it will always return the same result when called repeatedly. It does not effect the database objects.
   3. *What evidence, if any, do you see of the commonly-cited* impedance mismatch *problem in object-relational information systems?* When object oriented languages are in communication with SQL, there is a “mismatch problem” due to the fact that SQL cannot apply data that is encapsulated in an object.
   4. *Would the web service built in the lab be susceptible to SQL-injection attacks?* It would not be susceptible to SQL-injection attacks because it requires user authentication (username and password) for the database.
2. 1. The “Knight Rank” team’s Trello board indicates regular activity and is well organized. There are plenty of user stories, lots of helpful screenshots, and lots of apparent communication. All group members seem to contribute to it often although it does seem like Joe has been the primary lead in all the technical work. I think they are entering their time wrong still, making the second number the total time rather than anticipated time remaining. Overall, they have a well-organized and consistent system.
   2. The database structure is logical and not overly complicated. In their presentation, they did not indicate on the schema which relationships were many to many or one to many. Also, in their “elo rating system” formula, I interpreted a need for historical scores to calculate the “expected” ranking value. However, there is no value allocated to storing score history in the SportRanking table (or anywhere else). Everything that was implemented is clear and was well explained in the presentation.
   3. Although the UI is not the most attractive, I am impressed by the progress of this group. They have their database connection successfully implemented and have even added some extra fun quirks like the app color selector. I didn’t really like the fact that users couldn’t add their own activities to the list but if it is pre-populated with a general list of popular activities it might be ok. There seems to be a lot of hard work put into coding this app and it is coming along well.
   4. Their presentation could have done with more passion to really sell the product, but they still did convey the purpose clearly. Again, I was impressed that they had the database implemented during the demonstration. They had good answers to the audience’s questions and acknowledged what they would or would not have time to reasonably complete.