1. I identified the work on the models

Director – this is new

Movie – Added

- DirectorId

- DirectorName (last, first),

MovieRating – added

* MovieId

Not only is model work a logical first place to start but if you are working with a separate team doing the UI work they could get started with your model definitions

1. I created the IDirectorRepository interface and the DirectorRepository class
2. I added the DirectorRepository to the RepositoryFactory class
3. I updated the IMovieRatingRepository interface and MovieRatingRepository class - added

* GetAverageRating method
* GetRatingCount method

1. I updated the IMovieRepository interface and MovieRepository class – added

* Private SetDirectory – you have to create an instance of DirectoryRepository for this
* Private SetAverageRating – you have to create an instance of MovieRatingRepository for this
* Private SetRatingCount – you have to create an instance of MovieRatingRepository for this
* In your GetMovies method also set up the director and movie rating information for each movie
* In your GetMovieById method also set up the director and movie rating information

From this point you should have the data you need for your Views and Controller work.