The project can be divided into a structure with three interconnected particles:

1. The frontend, which includes the user interface elements:
   1. The menus and buttons.
   2. Brush color and size selectors.
   3. The wrapping frame and static elements on it
   4. A tutorial for the player.
   5. Objects (nodes, edges, and if simulated the trucks)
2. The mid-backend. This part should be based on Aybike's calculations.
   1. The instantiation
      1. layout of the nodes and edges from a set of coordinates. This part is experimentally done already and looks a good flexible way for the future, to load information from other geographic locations. It should interact with the frontend elements.
      2. Saved data, to be loaded from DB (see 2.e)
   2. The calculations. It is basically a simulation of trucks moving based on the form of the brushed elements.
   3. Posting data for every single user-interaction to the backend server for future analysis.
   4. Record the user-interaction such that it can be replicated.
   5. Replication of User interaction from the stored back end DB
   6. Providing a continues connection with online backend DB to store (1. user interaction. 2. Saved calculations 3. read coordinates for the layout).
3. Backend (PHP or Python)
   1. Authentication of the user by a unique id.
   2. Store the data sent from the mid-backend,
   3. Answer to the queries for the stored data.

My suggestion:

Some of the tasks of the mid-back end can be done in the backend to speed up coding. Such as simulation and calculations parts. The current calculations are in MATLAB which is pretty almost like python, perhaps it can help both Aybike and Erinc to handle this part, and error control it. It also reduces amount of C# code in the mid-backend.