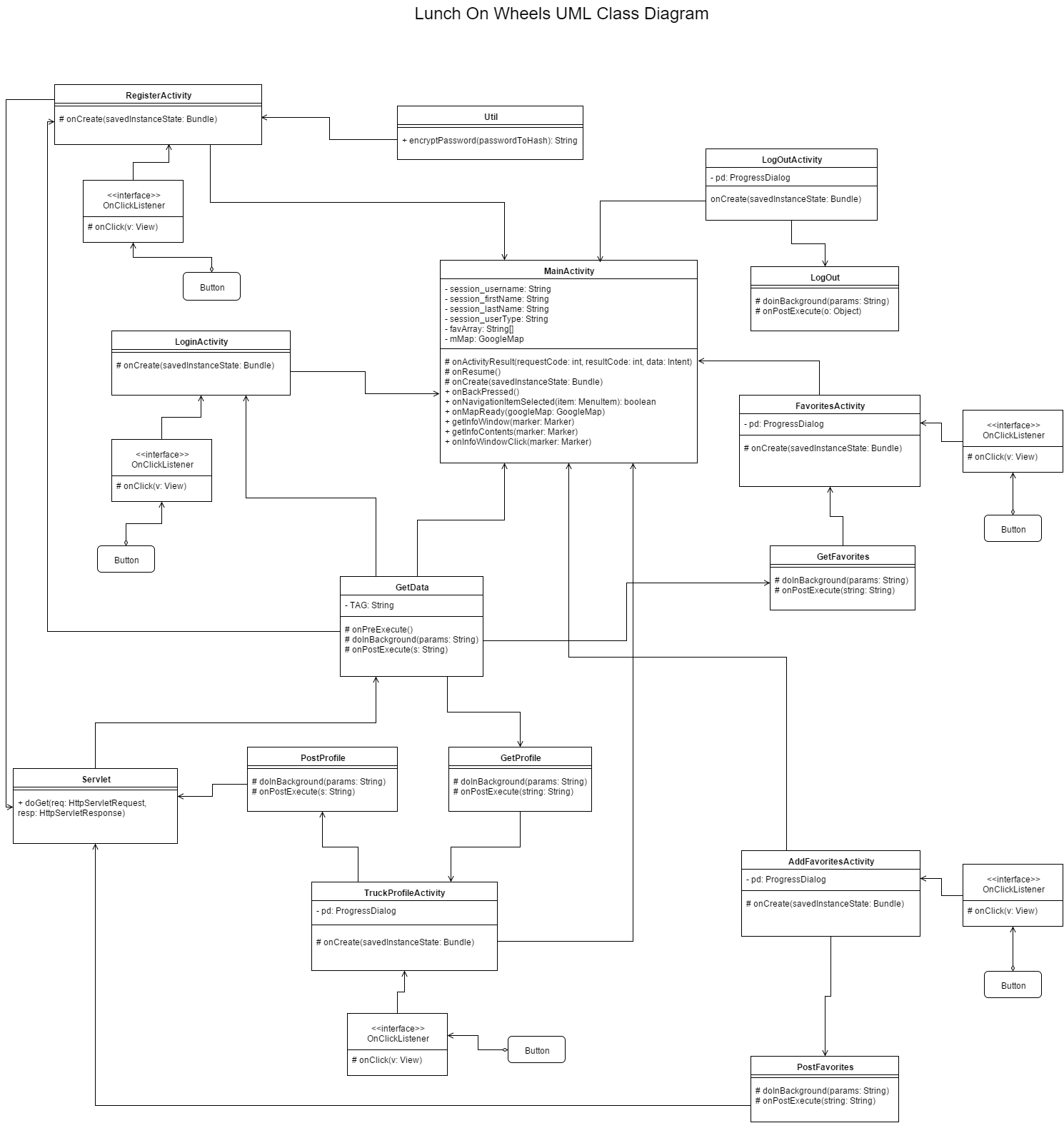
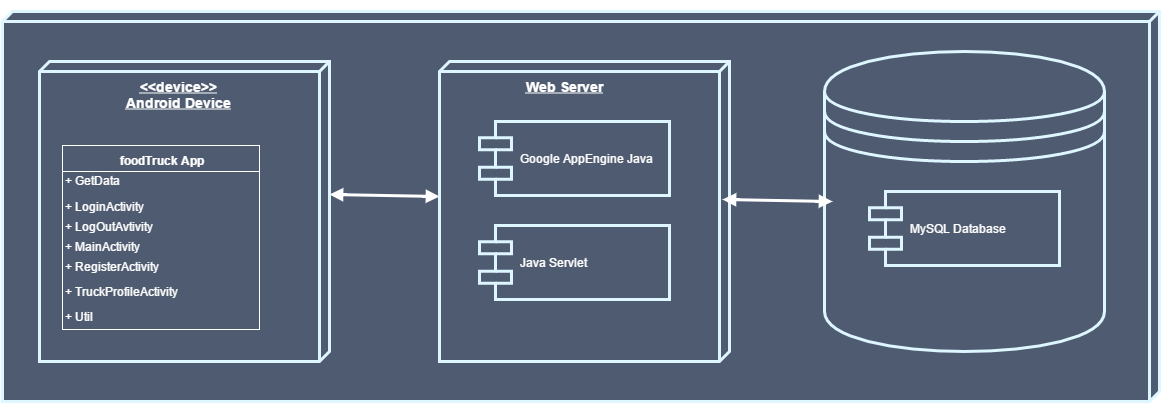
**Summary:**

Since Android activities are closely coupled to both interface and data access mechanisms, our team chose the [Model - View - ViewModel design pattern](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93viewmodel). The design pattern of the code was influenced by the structure inherent to software developed in the Android Studio framework.  The View is the user interface derived from both the layout xml files and the ViewModel java classes. The Model is the backend logic which consists of the Java servlet, and the ViewModel are the java classes that were written to communicate and modify the data, driving changes from the backend logic to the View.  The UML Class diagram shows the interaction of the ViewModel with the View and Model as well as the class relationships. The Deployment diagram and Database Entity-Relationship diagram shows the model. The UML Sequence diagram and overview Use Case diagram shows the interaction of the view with the ViewModel and Model.

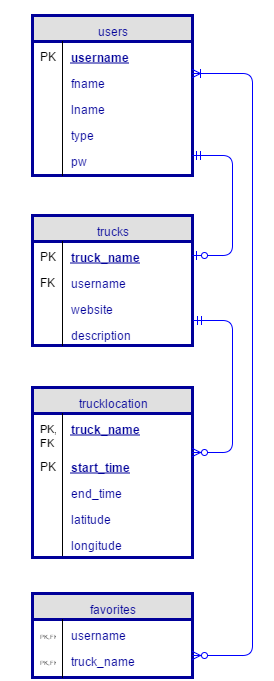
**UML Class Diagram:**

****

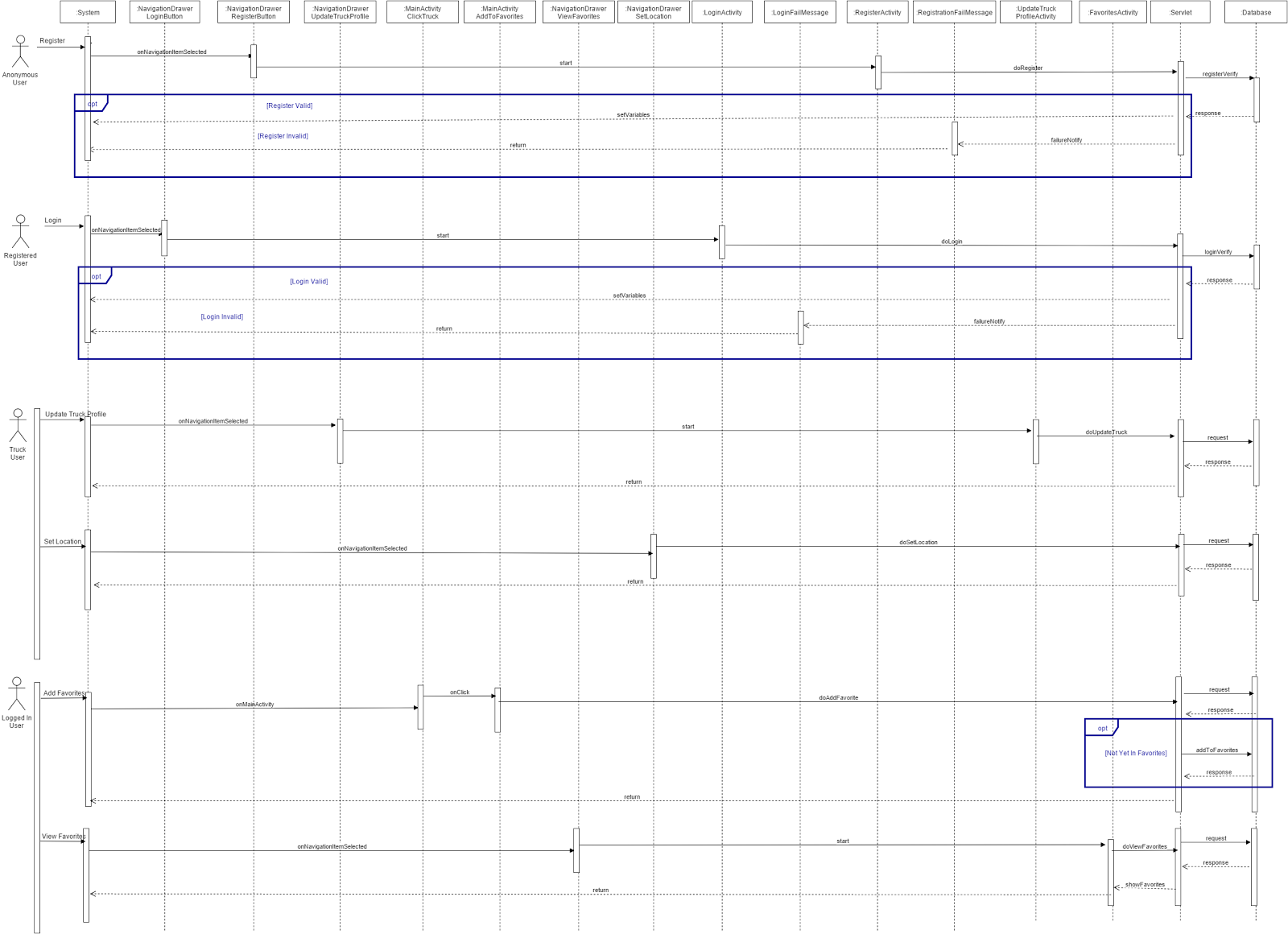
**Deployment Diagram:**

****

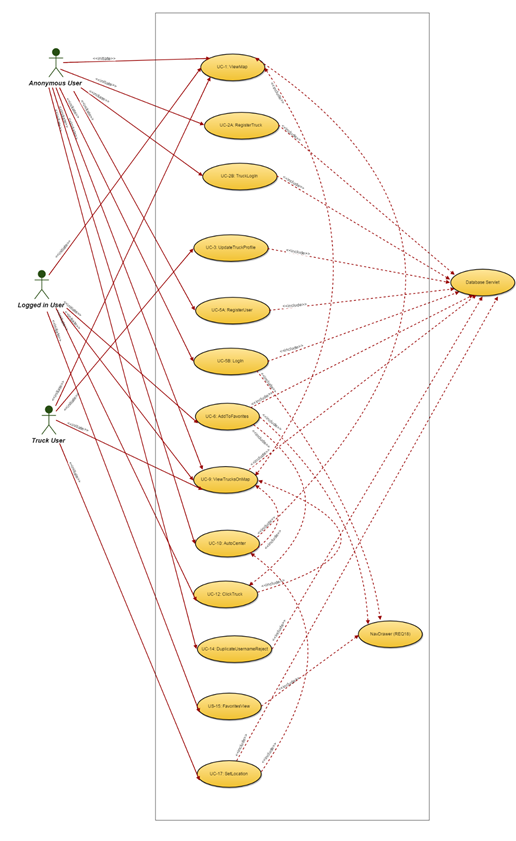
**Database Entity-Relation Diagram:**

****

**UML Sequence Diagram:**

****

**Use Case Diagram:**

****