**C Sc 335 Analysis and Design Artifacts for Final Project**

**1. Team Name:**  Localhost/127.0.0.1

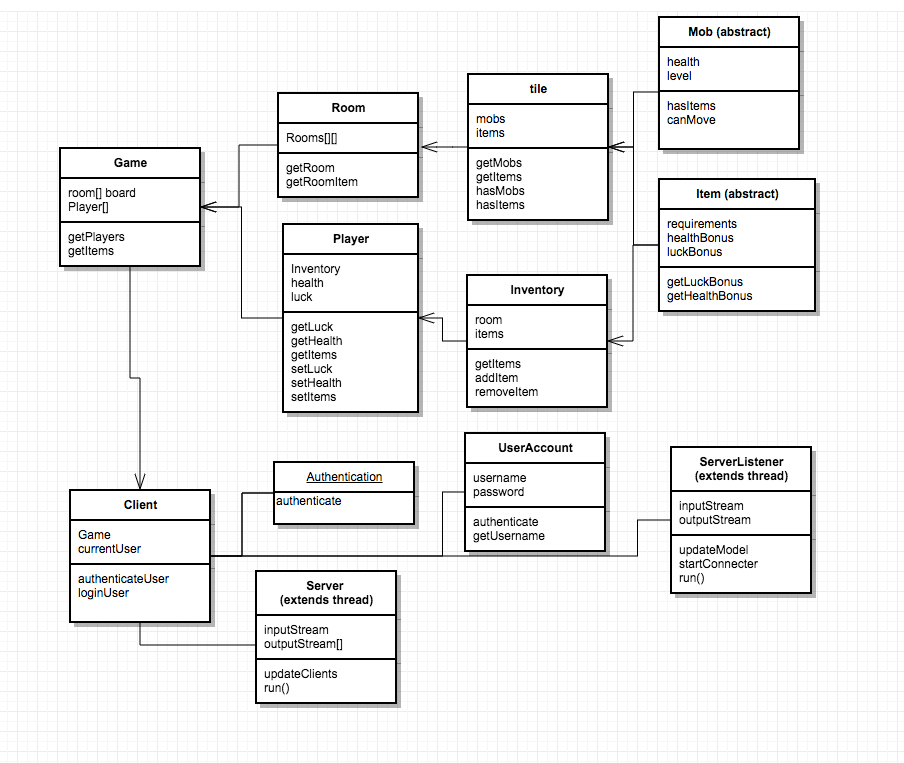
**2. Team Members**: Andrew Heyer, Brian Lovelace, Richie Ren, Chris Derck

**3. Candidate Objects or Class Hierarchies**

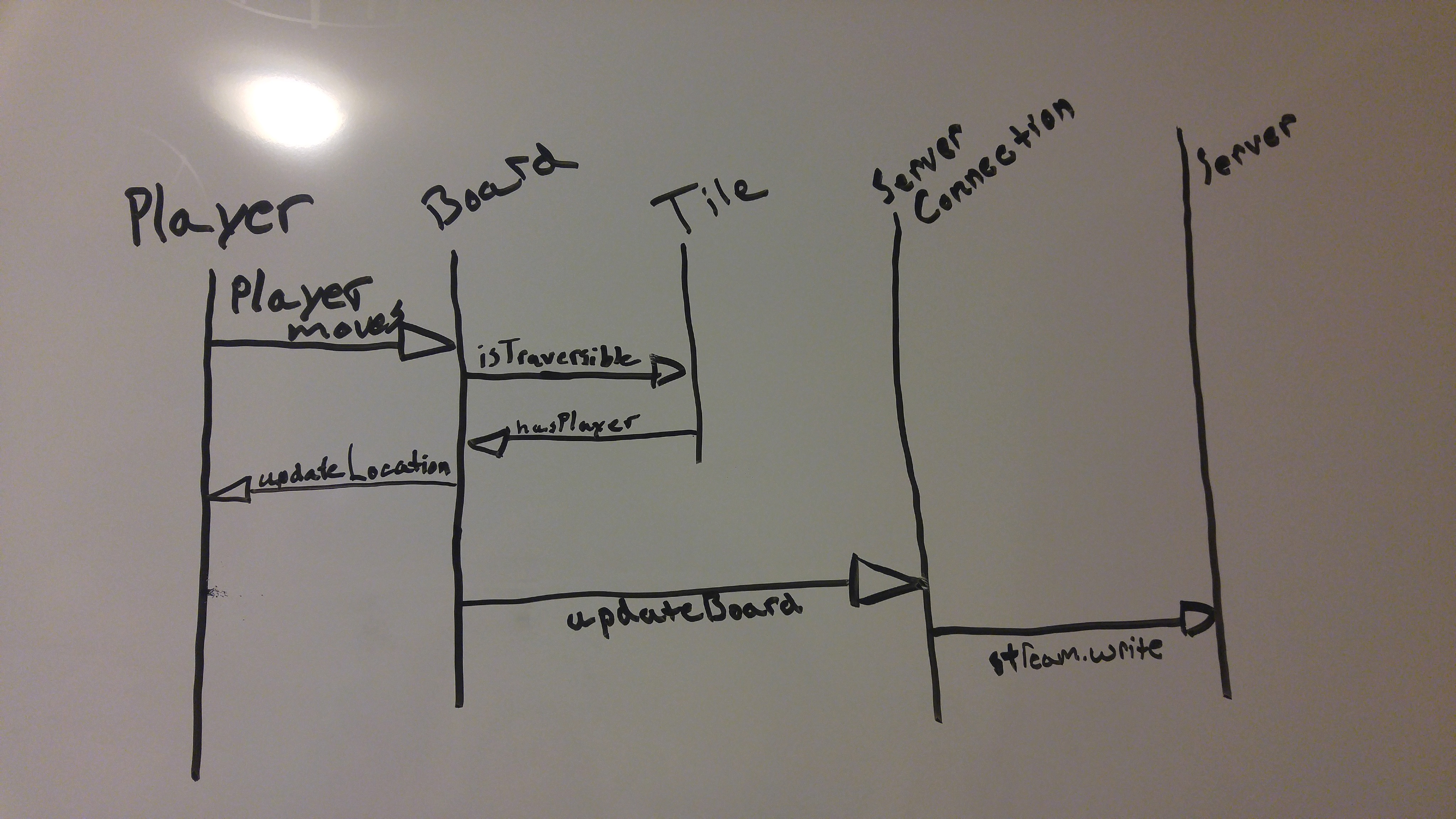
List the most important objects, or the name of an inheritance hierarchy, and the main responsibility.

|  |  |
| --- | --- |
| **Candidate Object** | **Single Responsibility in 1 or 2 sentences** |
| UserAccount | Store user specific information |
| Authentication | Checks the user login credentials |
| Server | Store the global game state. listen for clients actions, and updates client game |
| Model | Maintain game state |
| Player | Store players stats, gives actions to model and server |
| Inventory | Stores items held by the players |
| Mob (abstract) | Implements basic mob functionality |
| Item (abstract) | models behaviors that all items wil have |
| ServerListener | Listens to the server to update all clients |
| Room | models behavior that the rooms will have |

**4. Class Diagram:** Your team UML Class Diagram must show at least all of your candidate objects from above. Show any relationships between them the classes such as inheritance or interface implementation. Draw general associations such as dependency or aggregation. Label some to help explain things. Add any multiplicity adornments that seem appropriate. Use notes to explain things if you feel it will help. Each UML class must show the class name. For full credit, each class must have an average of at least one attribute per class. There must be an average of at least 2.0 methods per class, which may be implicit (no need to repeat methods) if the class implements a Java interface with methods shown there.

**

**5. Sequence Diagram:** Your team UML Sequence Diagram should show the most important scenario you can think of. Your sequence diagram should show most of your objects from above and how they communicate with each other.

**