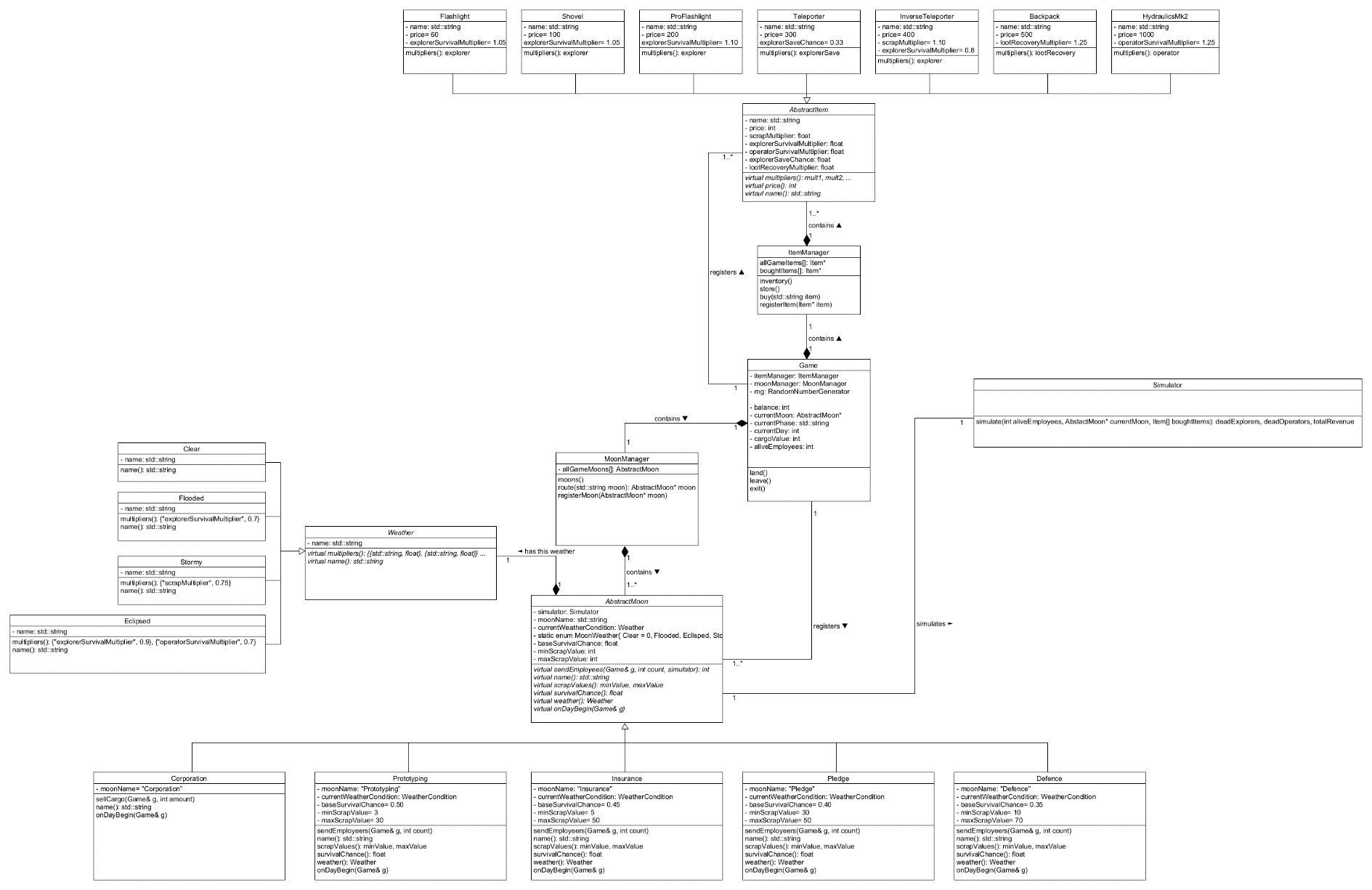
C++ A1 Design Document

Miguel Angelo Bautista [baumy007@mymail.unisa.edu.au](mailto:baumy007@mymail.unisa.edu.au)

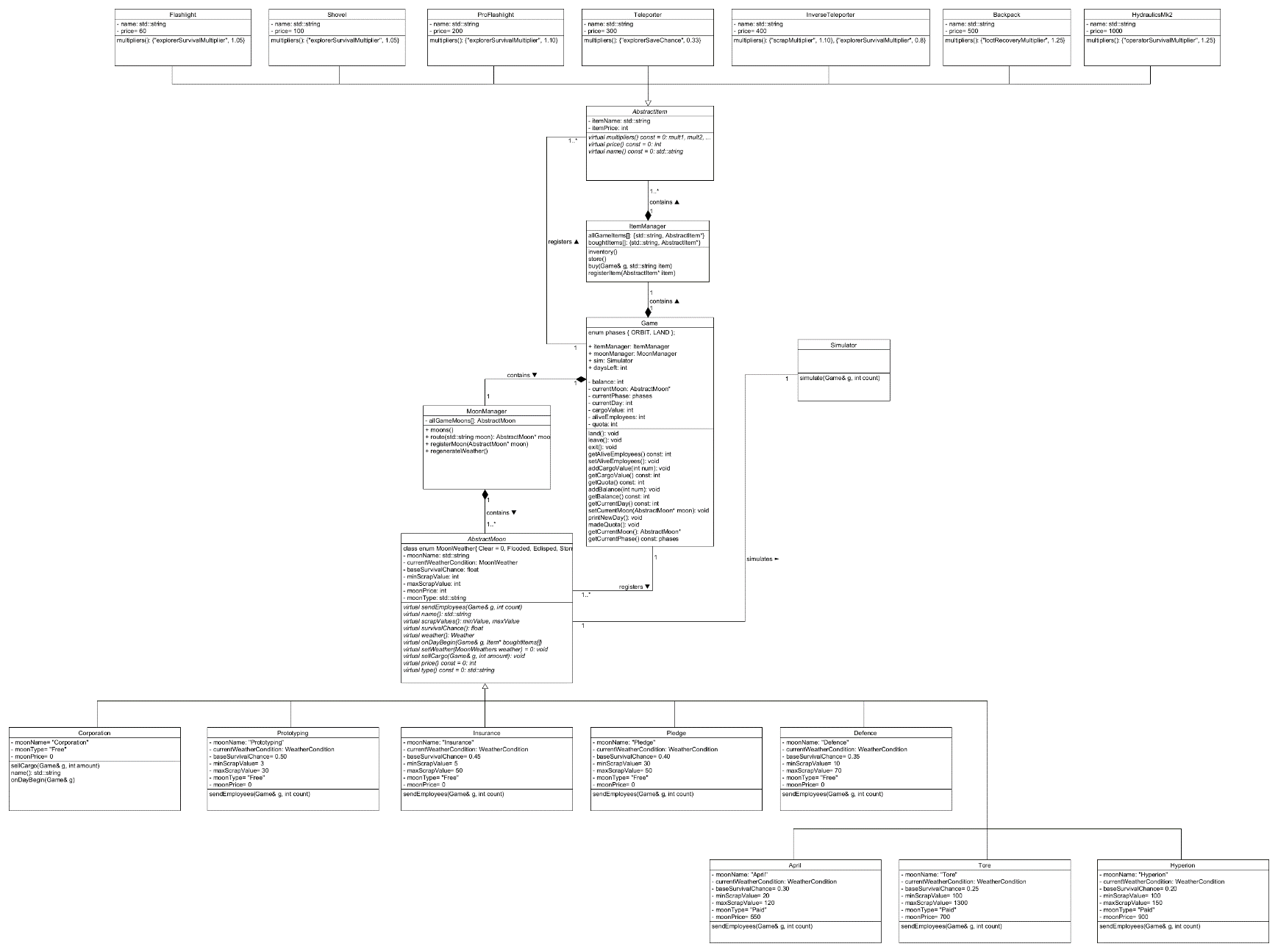
110355545

NEXT PAGE ---->

UML Diagram Original:



UML Diagram Updated 12/04/2024



Design Decisions

* Weather classes were removed and all multipliers are stored in the Simulator
* daysLeft in Game class is public to allow a small fix in the days left calculation
* MoonManager is in charge of regenerating each moon’s weather as it has all the moons stored in it
* Each item’s multiplier() returns an unordered\_map of all the multipliers set to 1.0 (no change) except for each item’s modifying multiplier.
* printNewDay() allows main to print out all the new day’s details such as current cargoValue, orbiting moon, balance, etc.
* madeQuota() is called by moon whenever quota has been reached. It increases the new quota and prints those details out
* addCargoValue() and addBalance() in Game class instead of a setter for more safer value changes
* scrapValues() in Moons returns a tuple minScrap, maxScrap used in the simulator
* Simulator’s simulate() taking in a Game& g argument in order to lessen the argument count