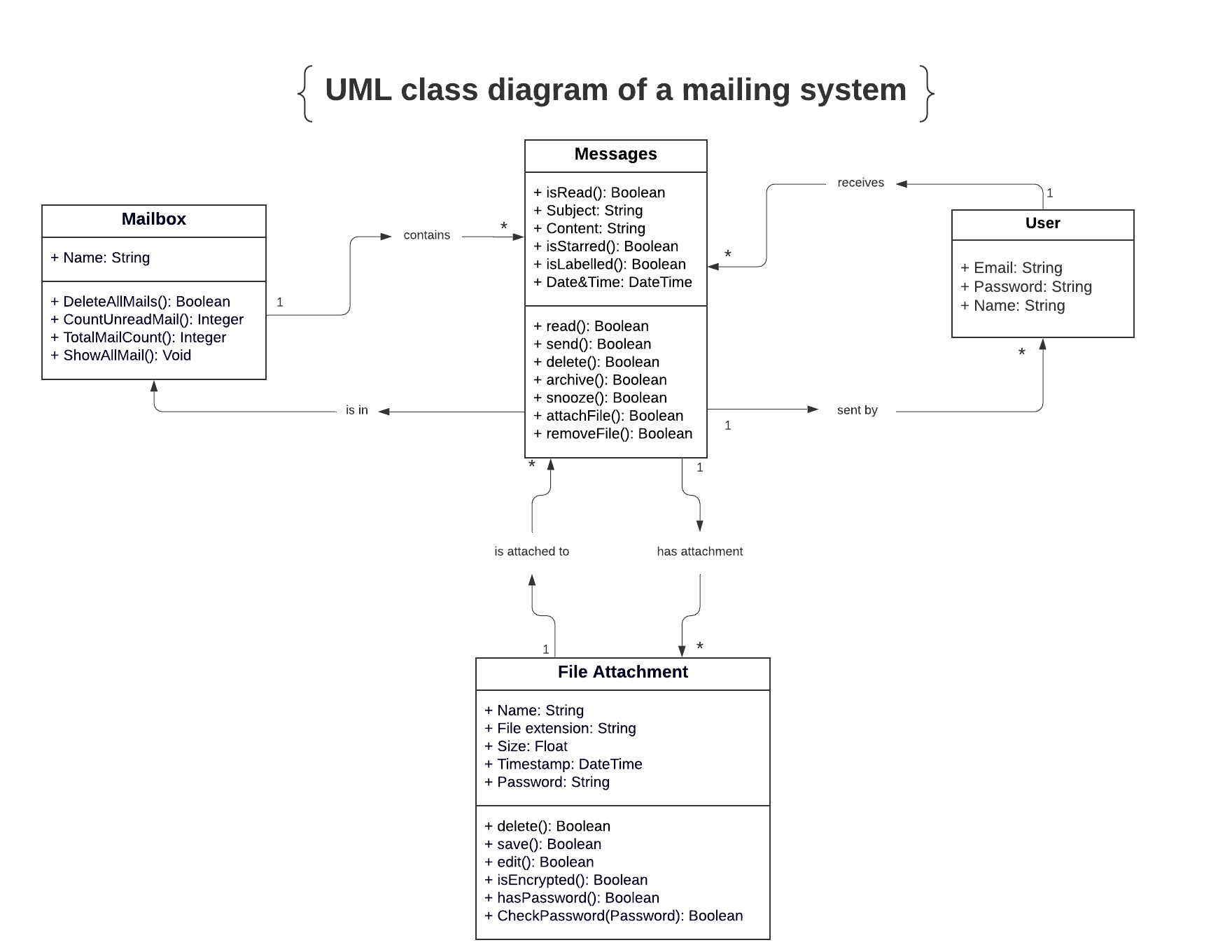
1. Explanations of concepts:

* Generalization: A technique for managing complexity by grouping entities with common characteristics in more general classes. We can examine these shared characteristics and avoid learning entities individually.
* Aggregation: In real-life, collection classes are generally composed of many diﬀerent smaller parts or components. Aggregation is a special type of relationship in which objects are combined to form bigger and more complex objects.
* Activity diagram: Activity diagrams are a type of diagrams in UML which can be used for the essentials of a system. It is a representation showing the workﬂow activities and actions involved in a process.
* State diagram: State diagram is a type of diagrams in UML which can be used to represent the reactions of a system to external and internal stimuli. This type of representation is important since knowledge of how a system reacts to its surrounding helps in developing a better system.
* Design pattern: Pattern is a description of a problem and its solutions in such a way that it can be reused in diﬀerent settings. It is a method of reusing knowledge and experience and is generally associated with object-oriented design.
* Host-target development: Host-target development is a kind of software development technique in which the software is developed on a host machine and runs on a target machine. Thus the software being developed must have the capability to run on diﬀerent platforms.

2. UML class diagram: