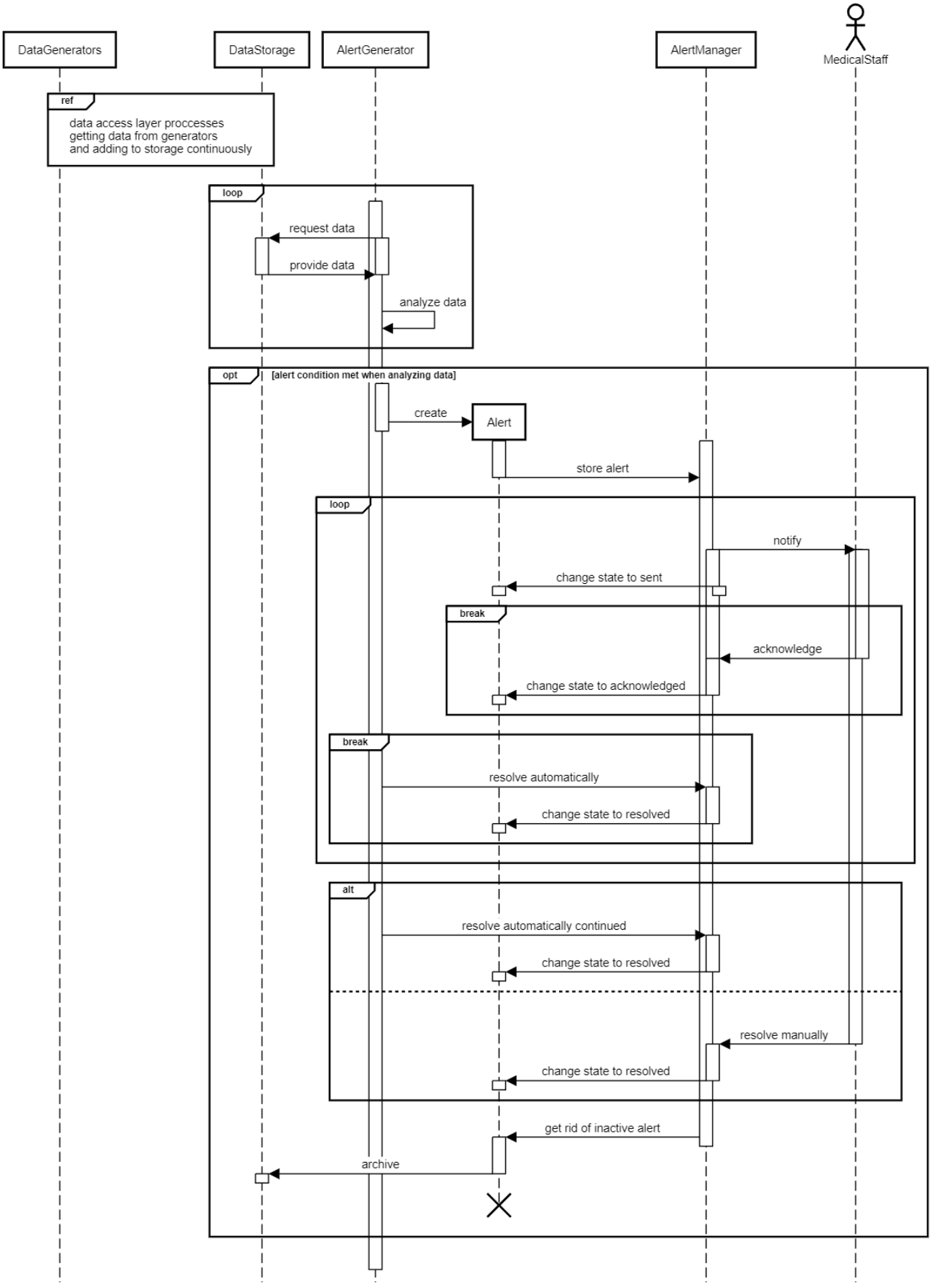
Alerting process sequence diagram



First, the data is created in data generators, processed in the data access layer and ends up in DataStorage. From there it is continuously pulled and analyzed, hence the infinite loop in the top section. I represented the constant analyzing to both create and disable alerts with a continuous execution block. If the condition for triggering an alert is met, the optional block is triggered, where an alert is created and then supervised by AlertManager. AlertManager acts as the bridge between the MedicalStaff and the Alert, waiting for the staff to react to the alerts and interacting with alerts accordingly. The loop continuously notifies the staff until the alert has been noticed, which triggers the first break option, or it resolves automatically, which triggers the second break block. It should be noted that AlertManager should not try to update the alert's status with every update and instead do it once only – this is a design flaw on my part. The alternatives box shows two ways how an alert can go from an acknowledged to resolved state. I was unsure how to best represent that automatic resolving can occur both before and after acknowledgement from staff.