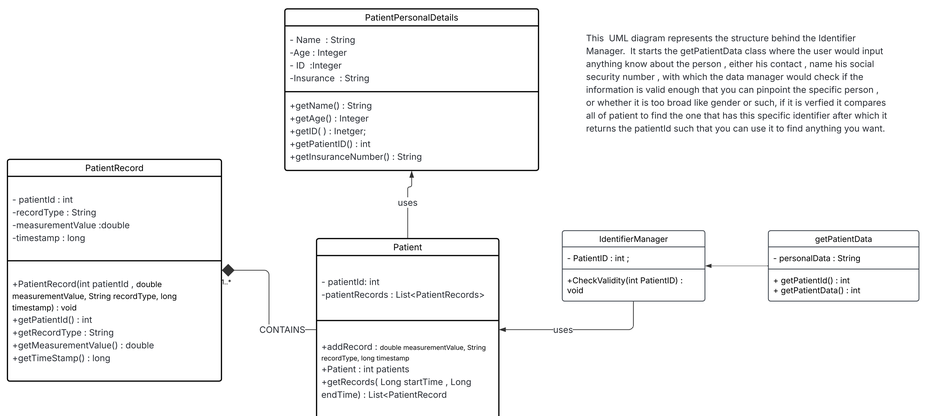
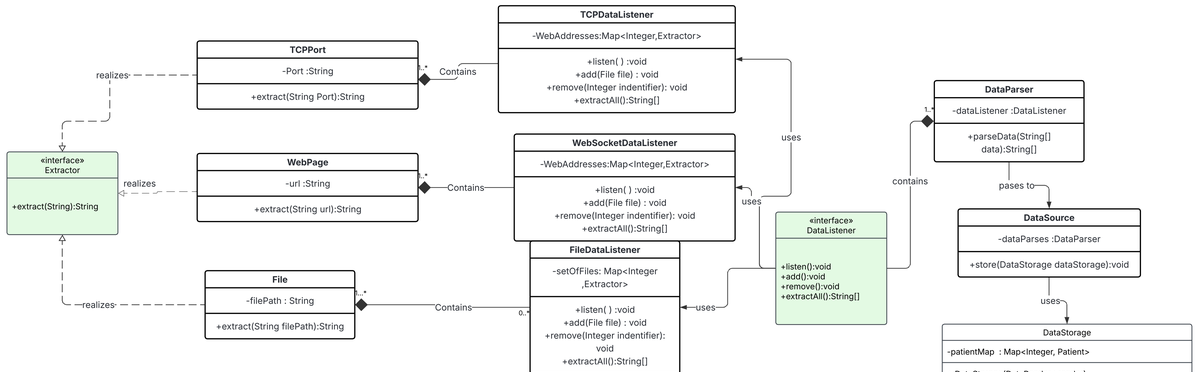


This UML diagram represents the structure behind the Identifier Manager. It starts the getPatientData class where the user would input anything know about the person , either his contact , name his social security number , with which the data manager would check if the information is valid enough that you can pinpoint the specific person , or whether it is too broad like gender or such, if it is verfied it compares all of patient to find the one that has this specific identifier after which it returns the patientId such that you can use it to find anything you want.



This uml diagram creates a system that can listen to incoming data from a variety of sources such that it can then be stored and saved for future use. The program originates at the DataSource where we save the incoming data into the storage class. Naturally before the data is being saved and stored it needs to be filtered and standardised, which is why we have a DataParser class which take in a Interface such that we can use the same method for all three types of input. These inputs are TCP, WebPages and Files. Indeed, from a hospital we can expect there to be more than one incoming file, webpage or server , which is why the DataListener classes have their own corresponding Class which allows us to extract individual files as well as groups or totality of files. These individual files web pages and files are saved in their own data type