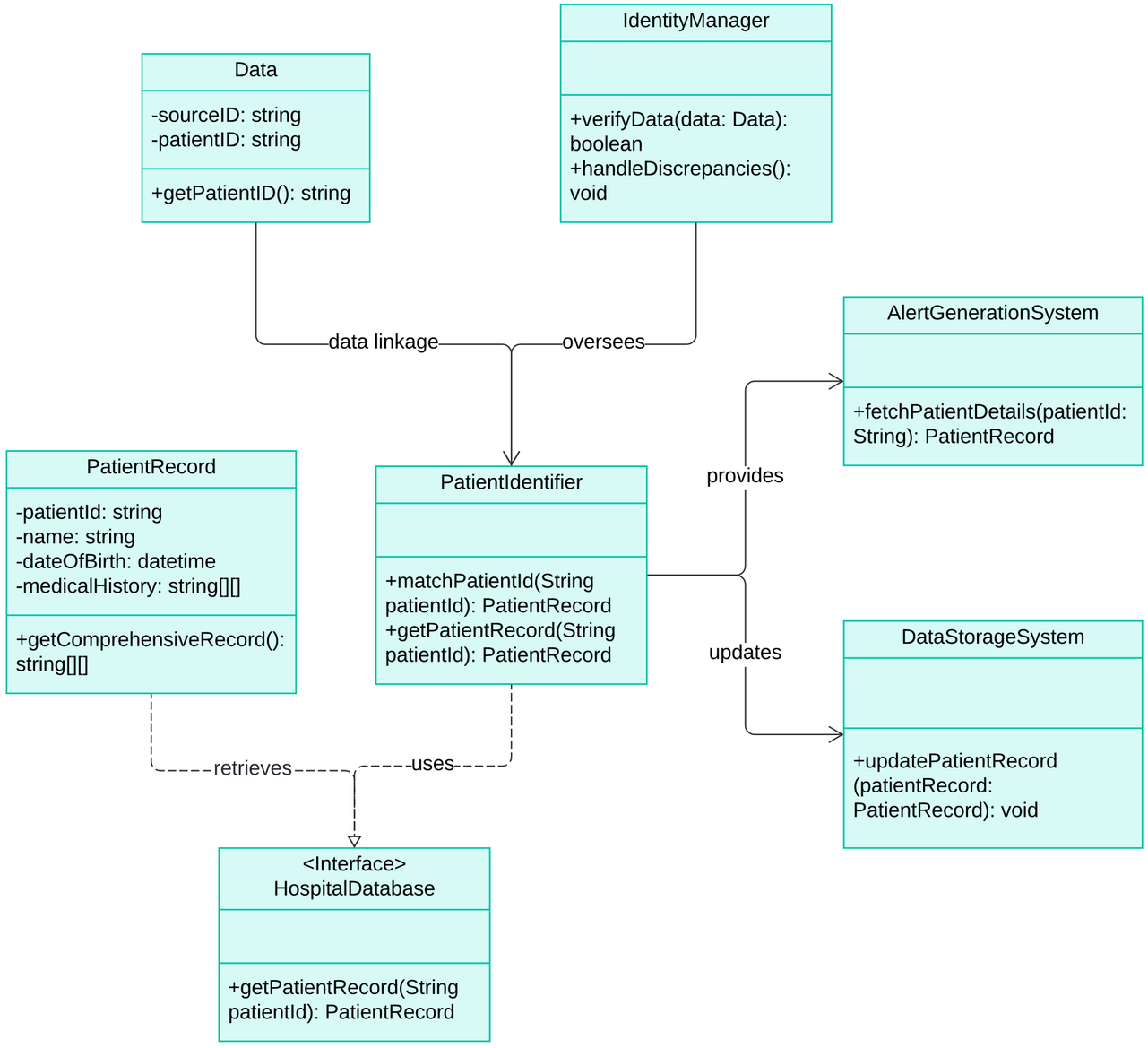

1. **Data**: Serves as the base data structure with source and patient identifiers, implying a system that handles data from multiple sources.
2. **PatientRecord**: Contains detailed patient information, such as ID, name, date of birth, and medical history. It provides a comprehensive record, indicating that the system is designed to give a complete overview of the patient's medical background.
3. **IdentityManager**: Oversees the integrity of data with functions to verify data and handle discrepancies. The "oversees" arrow to Data suggests that it plays a critical role in ensuring the accuracy and consistency of the patient data flowing into the system.
4. **PatientIdentifier**: Acts as a service to match patient IDs to their records, indicating a system with robust data matching capabilities to ensure the right data is associated with the right patient.
5. **AlertGenerationSystem**: Uses patient details to possibly generate alerts, with a direct link to PatientRecord indicating it accesses detailed patient data to perform its function.
6. **DataStorageSystem**: It updates the patient records, signifying a system designed to keep patient data current and accurate.
7. **HospitalDatabase**: Presented as an interface, it stipulates contracts for retrieving patient records. The dashed line to PatientRecord and PatientIdentifier suggests that both classes implement this interface, ensuring they can provide the necessary data retrieval functionality.

The diagram reflects a design focused on data integrity and patient-specific alert generation. It shows a layered approach where data is validated and managed before being used to generate alerts, which is crucial in healthcare where accurate and timely information can have significant implications.