Blockchain-Based Cooperative Medical Records Management System

(System Design)

1. System Functionality

The system provides the different functionalities for each stakeholder in it through both smart contracts written in Solidity and blockchain characteristics. Figure 1 depicts those functions which major to meet the objectives of the project.

For instance, Healthcare provider can create Patient's medical record and insurance claim that must be processed by insurance company to be either approved or rejected. The doctor can update patient medical record. In addition, the patient has Access two main features in the system which are create access policies and view medical record contents any time need for that.
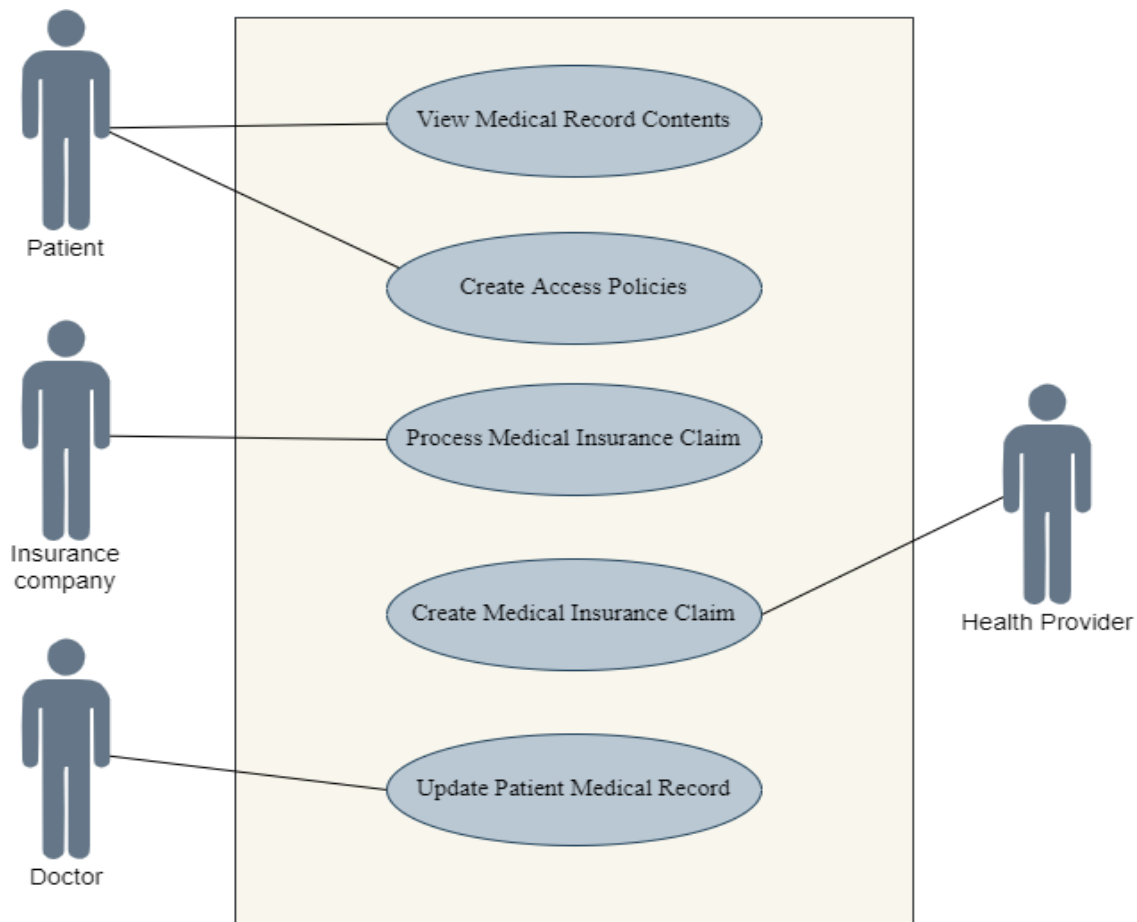

Figure 1: Auditing and Tracing Medical Record Use Case diagram.

2. System Workflow

The system workflow presents how the system works and depicts the interactions that occur among the various components. Figure 2 illustrates the big picture of system workflow, covering all the major transactions involved.
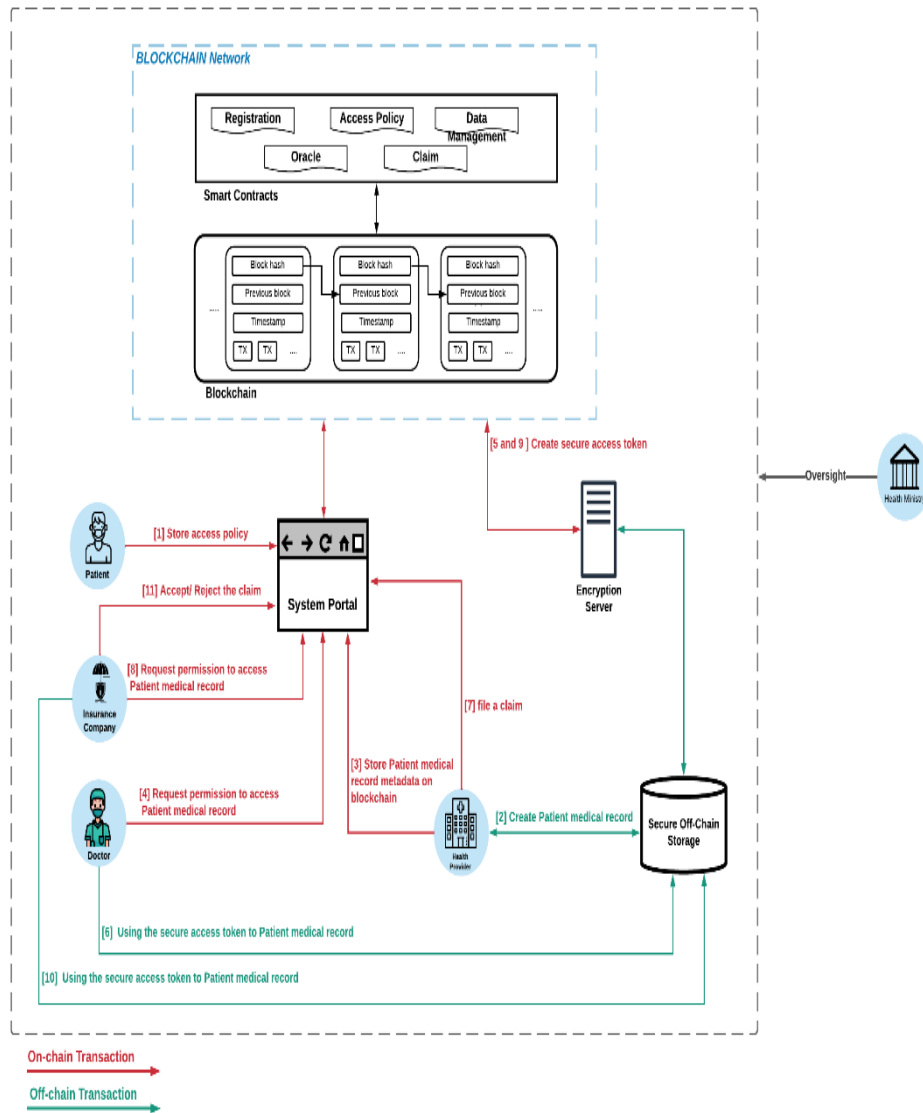

Figure 2: Auditing and Tracing Medical Record: Workflow Diagram

The steps involved in the workflow are as follows:

- o Step [1]: The patient stores access policies on the blockchain, which enables the system to evaluate access requests without the need to approach the patient directly. Access requests are automatically granted or rejected.
- o Step [2]: When the patient needs treatment, the healthcare provider creates the patient's medical record and stores it on the database.
- o Step [3]: The healthcare provider stores the reference to this record on the blockchain network in order to ensure the integrity of patient data and enable other parties in the system to access the medical record.
- o Step [4]: To complete the treatment process, the doctor requests permission to access the patient's record using the patient's address on the blockchain. The request is processed through the access policy in the smart contract, which considers the request and checks the patient's access policies.
- o Step [5],[9]: To grant access to the doctor/insurance company, the request is redirected to the encryption server in order to generate a secure access token for the doctor/insurance company. This process is completed using smart contract, Registration.
- o Step [6]: The doctor, using this secure access token, browses the patient's medical record and adds visit details.
- o Step [7]: In the event that a patient requires the insurance company's approval for a particular treatment plan and so on, the healthcare provider files the patient's insurance claim, which is stored in the smart contract on the blockchain network, Claim. Subsequently, the insurance company is notified.
- o Step [8]: The company requests permission to access the patient's record by employing the methods delineated in Step [4].
- o Step [10]: The healthcare provider uses the secure access token to access the patient's medical records.
- o Step [11]: The healthcare provider evaluates the claim made by its medical staff based on the data in the patient's medical record and then either accepts or rejects it.

The above-mentioned operations will be executed under the supervision of the Ministry of Health in the Kingdom of Saudi Arabia, thus ensuring their integrity.

The following diagrams show the detailed workflow of each feature in the system:

2.1. Creating/updating Access Policies Workflow

In order to add/update access policies of patient, the patient have to login the system then add/update the access policies those want. The access policy is stored in the system which used later to check the accessibility of a patient's medical record on the blockchain network. Figure 3 presents workflow of create – update access policies.

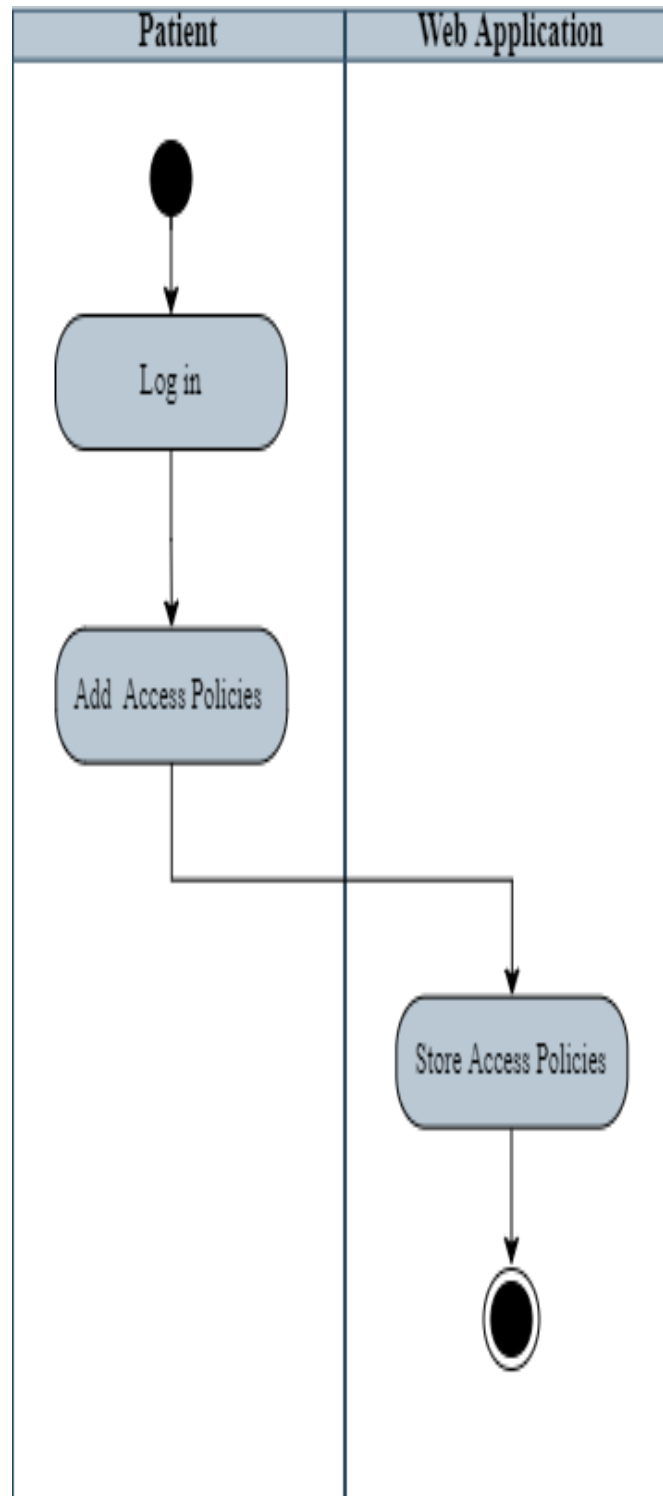| Patient	Web Application

Figure 3: Creating – Updating Access Policies: Workflow Diagram

2.2. Viewing Medical Record Workflow

As shown in Figure 4, the workflow regarding the viewing of medical record by the patient starts with the patient logging into the system. Next, the system retrieves the medical record that the patient has asked for, from the database on the blockchain network, and shows it to the patient. Finally, the patient is able to view and browse all contents of the record.
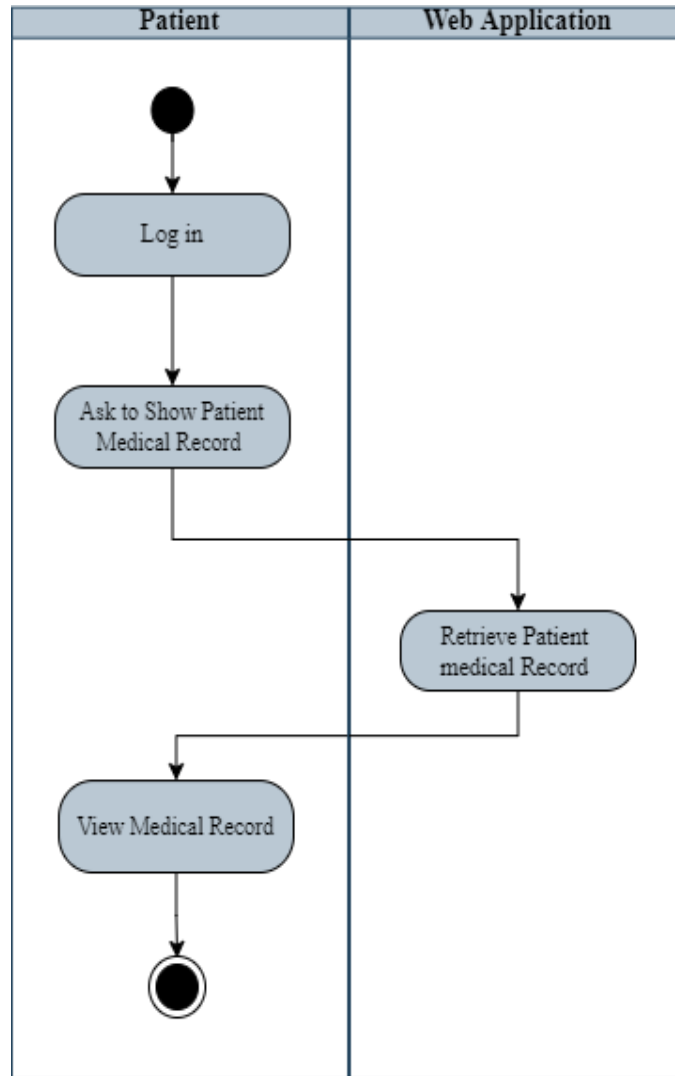

Figure 4: Viewing Medical Record: Workflow Diagram

2.3. Creating Medical Insurance Claim Workflow

Steps involved in creating medical insurance claim are shown in Figure 5, wherein the health care provider creates a medical insurance claim upon the patient's request. Next, the medical insurance claim is stored on the blockchain network, so that the insurance company can receive and process it later.
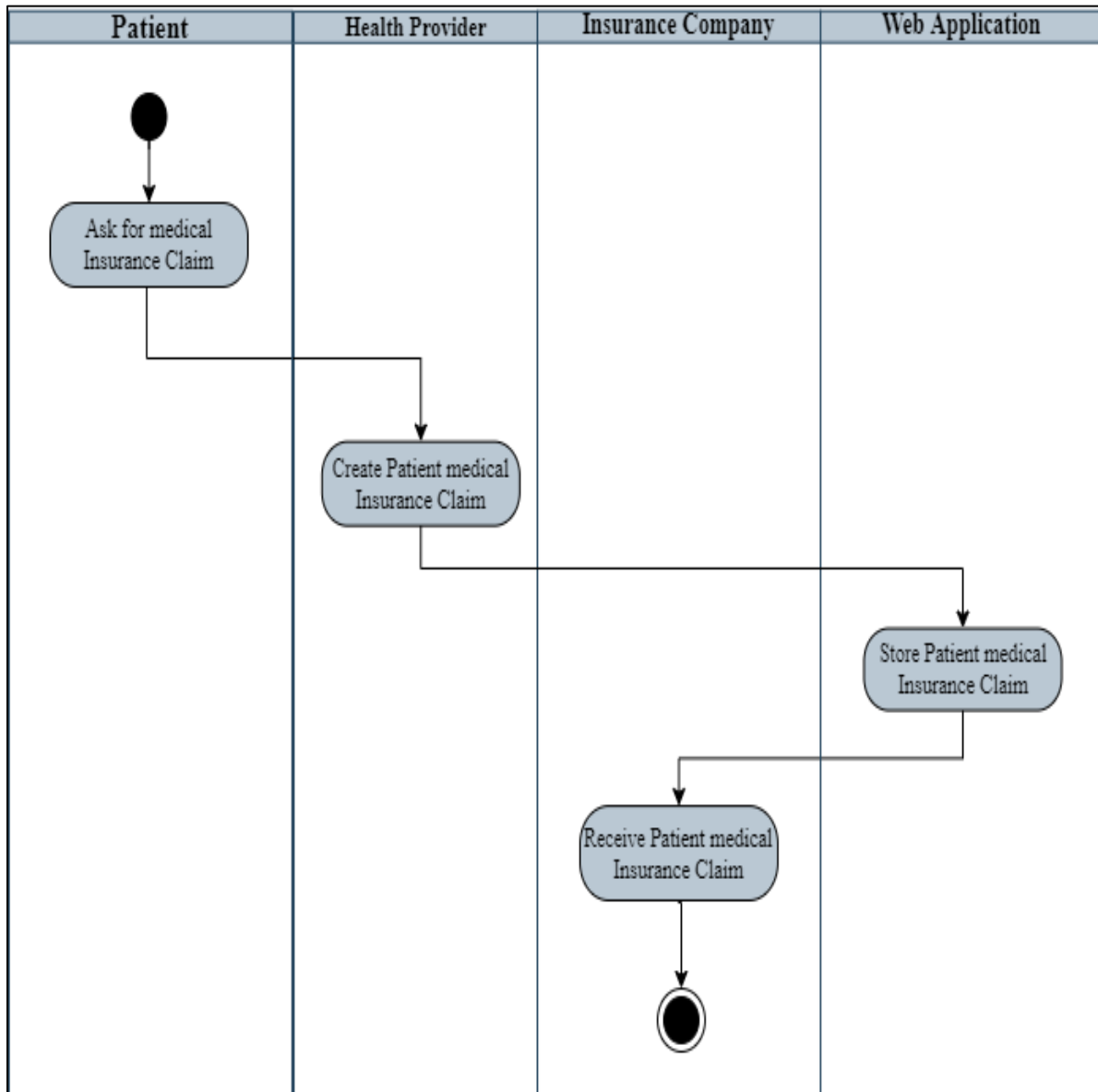

Figure 5: Creating Medical Insurance Claim: Workflow Diagram

2.4. Processing Medical Insurance Claim Workflow

Once the insurance company receives insurance claim for the patient, it requests access to the patient's medical record in order to process the claim. The system checks the access polices for the patient, so based on that, insurance company will reject or grant permission. Once the insurance company has access to the medical record, it processes the claim and adds the result (accepted or rejected). Next, the system updates the patient's medical record on the blockchain network and notifies the patient. Figure 6 illustrates the workflow related to processing the medical insurance claim.
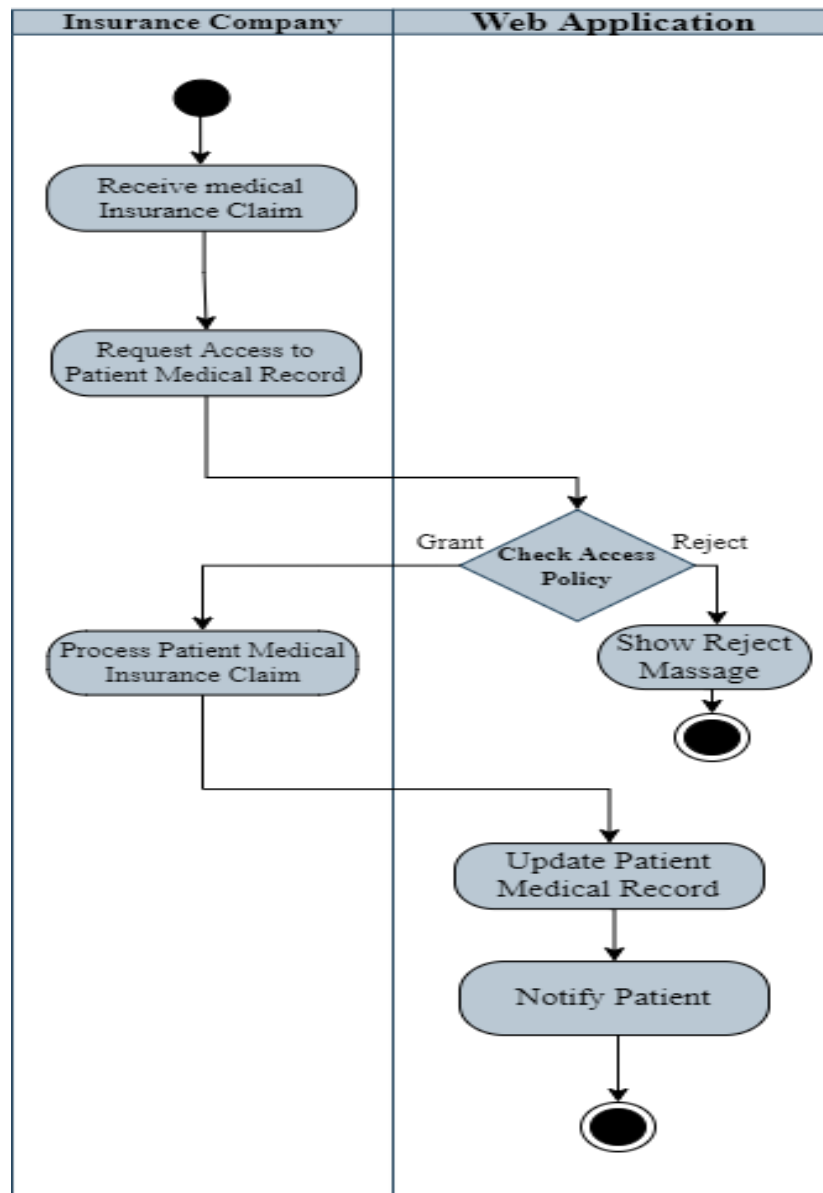

Figure 6: Processing Medical Insurance Claim: Workflow Diagram

2.5. Updating patient's Medical Record Workflow

As shown in Figure 7, the workflow of updating the patient's medical record by the doctor starts with the patient requesting a treatment. Next, the doctor requests permission to access the patient's medical record. The system checks the access polices for the patient, based on which the doctor grants or rejects permission. If permission is granted, the doctor can access the medical record, review it and add a summary of the medical visit. Conversely, if access denied, the system sends a rejection message to the doctor.
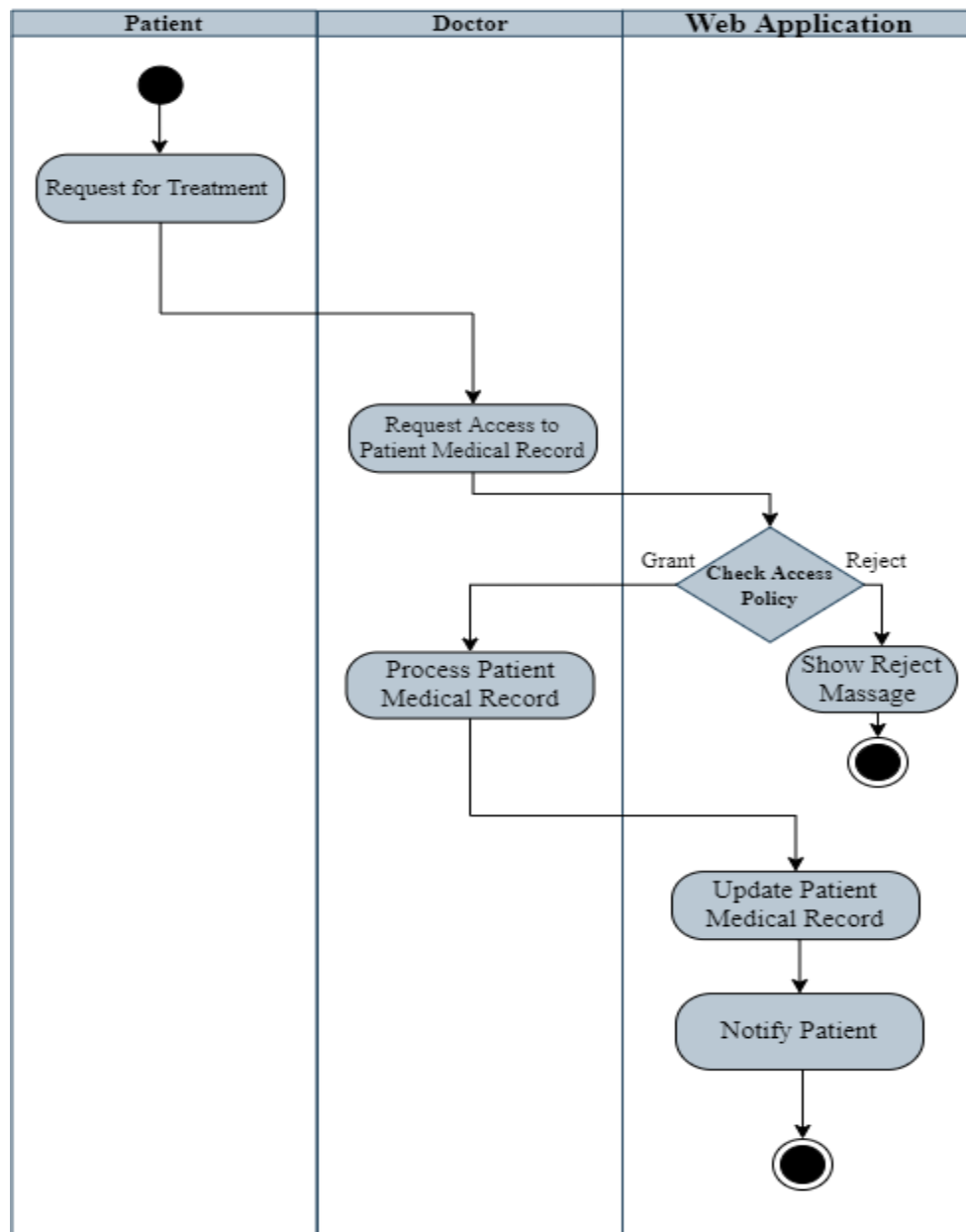

Figure 7: Updating Medical Record: Workflow Diagram

3. Entity Relationship Diagram (ERD)

To build a database for DApp system architecture, MongoDB—a flexible NoSQL database platform—is one of the most suitable storage databases. MongoDB allows the data's structure to be changed over time. It has strong features that solve the problems and limitations of relational databases by ensuring high availability and a high level of performance and providing a dynamic schema. In our system, MongoDB has been used to store off-chain data, which means store data out of blocks and take only the reference of data to be in the blocks; this provides additional security and privacy.

In contrast to relational and other NoSQL databases, MongoDB deals with collections and documents, rather than tables. The system comprises four collections: patient, doctor, healthcare provider and insurance company. For a patient, ID, name, date of birth, email address and gender are collected and stored; for a doctor, ID, name, email address, license number and hospital wallet address are collected and stored; for a health-care provider, ID, name, license number, email address, address and access point are collected and stored; and for an insurance company, ID, name, email address, commercial registration number and address are collected and stored. Figure 8 illustrates an ERD showing how data are saved in MongoDB.
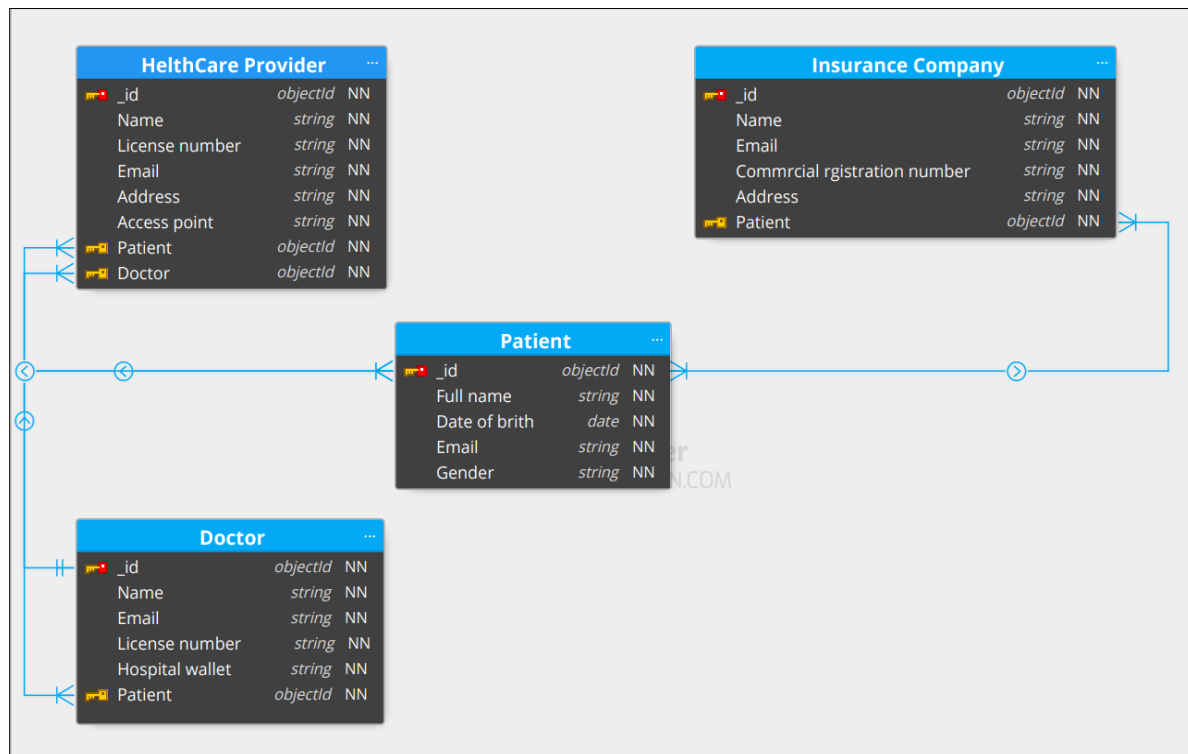

Figure 8: ER Diagram.