

Developing a Cloud-Based Healthcare Application

You are required to develop a cloud-based application/service for healthcare data storage, update, and retrieval. The application consists of the following components:

1. **Patient component:** Patients are the end users of the application.
 - 1.1. A valid patient must subscribe first to use all the features of the application.
 - 1.2. The patient needs to enter necessary information and upload documents for registration in order to subscribe.
 - 1.3. He can upload lab as well as medical images such CT scans, medical videos, etc. so that his doctor can see and check them.
 - 1.4. A patient can select one or more doctors based on his illness/needed medical advice.
 - 1.5. He can either choose to make an appointment with a doctor or make a follow up with the doctor. In making appointment the application must resolve any conflicts (e. g., If two patients reserving with a doctor at the same time).
2. **Administrator component:** The administrator is the person who has a medical back-ground. The functions of an administrator are as follows:
 - 2.1. Generation of unique patient ID: After verifying all the information and documents of a new user, the administrator registers the user as an authorized user and sends a unique patient ID for future login.
 - 2.2. Authentication with patient ID: Whenever the patient wants to login again, the administrator first verifies the entered patient ID.
 - 2.3. Update the data in cloud: The patient or the doctor cannot update or perform any modification on the data stored in the cloud. Only the administrator has the rights to update the cloud's data to attain data security. However, a patient can update his own information and documentation as needed or as requested by his doctor.

- 2.4. Account management: All registered user' (patients and doctors) accounts are managed by the administrator, who has the backup of all information related to a patient's account.
- 2.5. Billing issues: For accessing the cloud, the customer or patient has to pay charges. All billing issues are handled by the administrator.
- 3. **System component:** Cloud system stores patients' records, performs computation and any needed functionality such as patient-doctor appointment management, notifications, emailing, searching, advertising such as pop-up general medical advices.
- 4. **Doctor component:** Doctors are important users of the application.
 - 4.1. A valid doctor must subscribe first to use all the features of the application.
 - 4.2. A doctor needs to send necessary information for registration.
 - 4.3. Doctors are classified based on their specialty (e. g., Dermatologist, Anesthesiologist, Ophthalmologist, Pediatricist, Nephrologist, Psychiatrist, Pathology, etc.)
 - 4.4. A doctor manages his own patients, e.g., accepts and makes up appointments with his patients, follows up his patients. He also can see a list of his patients such as new patients, follow up patients, etc.
 - 4.5. Gives diagnosis, treatments and prescriptions to a patient. These have to be stored in the cloud and also sent to the patient.
 - 4.6. Sends notifications or emails to his patients as needed.

In developing the application, you must satisfy the following requirements:

- 1. The application should use a SQL and/or NoSQL database as necessary for storing and retrieving data.
- 2. As stated above the application must have appointment management, notifications, emailing, searching, advertising such as pop-up general medical advices and ads. The medical advices and ads have to be displayed on the doctor/patient screens while working in any of their screens.
- 3. All user activities in the system must be tracked and recorded by the system with date and time. This includes doctor as well as patient numbers and activities. This functionality is related to the administrator and partly to the doctor (for his patients).
- 4. (**Optional**) Based on requirement no. 3, the application must have its own analytics as necessary. This includes all activities performed by the admin, the doctor and the patient. All processing, storage, and main activities of the

application must be tracked through such analytics. Use the analytics tools of the selected PaaS.

5. The application must have a user friendly GUI with emphasis to each of its users (admin, doctor, and patient).
6. **(Optional)** Any cloud-based additions and improvements you might consider related to the application (This will worth additional credits).
7. You can use the platform (PaaS) of your choice and deploy it on the cloud of your choice.
8. The application must follow a well-defined and clear cloud-based software methodology. You have to document your development process adapting cloud oriented software development methodology showing your specification, design, architecture, implementation, and deployment to the cloud. The documentation must include a description of the used cloud platform (PaaS). it must include a user guide to access/use the application.

Note: Here, you can benefit from or use the methodology you have reviewed in Assignment#1: Report on cloud-based software methodology.

Submissions:

1. A cloud-based software application implementing the above requirements and any used packages and libraries. The application should be deployed in the cloud and in a software collaboration platform such as Github (or any of your choice). Include in the software development report:
 - 1.1. A link to the cloud application. It should be fully up and functioning.
 - 1.2. A link to the software collaboration platform such as Github. I need to see all the parts of the application including who of the group developed what of the corresponding parts/packages.
2. A software development report based on the last requirement (follow the attached template). It should contain explicitly the links to the Github platform, the link to the cloud where the application is deployed and runs, and the needed credentials (username and password) to access the application such as username and password. **Note:** Again, you can benefit from the report on the methodology you have reviewed in Assignment#1: Report on cloud-based software methodology.

Instructor: Dr. Rebhi S. Baraka

rbaraka@iugaza.edu.ps

<https://sites.google.com/iugaza.edu.ps/rbaraka/>