An analysis of how a medical centre should switch to cloud-based services

Joey Tatú  
15015556  
Cloud Platform Programming  
National College of IrelandDublin, Ireland  
10th January 2022  
joey.tatu@student.ncirl.ie

*Abstract*—This report looks at how a small medical centre could migrate to cloud-based services using Amazon Web Services. Considered cloud services are Amazon Lex to create an appointment system for the centre with an app on an Android tablet (console) to check in patients, a Debian 10 Buster (Linux) machine to store a 3CX phone system with Amazon Chime to port their current phone number, Amazon DynamoDB and Amazon S3 to use with patient accounts, vaccine records, prescriptions and scanned or uploaded documents, Amazon Rekognition to be installed on an app on an Android tablet at the main entrance to detect a patient’s temperature and if the patient is wearing a face mask and an Amazon EC2 instance with OpenVPN to create a virtual private network for use in the centre. The functional requirements are also discussed and how the cloud services are expected to work. Amazon Web services is only but one option and other cloud service provided should also be considered.

# Acronyms

|  |  |
| --- | --- |
| AWS | Amazon Web Services |
| EC2 | Amazon Elastic Compute Cloud |
| S3 | Amazon Simple Storage Service |
| 3CX | Developer of VoIP IPBX software on a global scale. |
| VoIP | Voice over Internet Protocol |
| IPBX | Internet Protocol private Branch Exchange |
| DB | Database |
| VPN | Virtual Private Network |
| GP | General Practitioner |
| NoSQL | Non-relational database |
| SaaS | Software-as-a-Services |

# Introduction

Did you know that you might be using cloud services without even knowing it? Most consumers just expect computer services and devices to work without even a second thought about what goes on in the background or in the cloud. A lot of services today are cloud-based services.

The main objective of this report is to analyse a small medical centre who are using manual and limited online tools and transfer these services to the cloud, reducing costs by freeing up time with staff and having faster services through the cloud.

Cloud computing boosts productivity while cutting costs in the healthcare industry. Cloud computing streamlines and protects medical data interchange, automates backend procedures, and even facilitates the creation and maintenance of telehealth apps. [1]

The health industry needs data privacy while adopting cloud services, which is a critical concern for patients. Environmental protection is crucial, and cloud-based services for the health industry include paperless documents and prescriptions. It's also important not to become overwhelmed by all the services available, as patient data should never take precedence over patient care. [1]

# Description of company

Walkinstown Medical Centre, located in Dublin 12, provides these services: GP, nurse, and chiropodist. [2] [3]

There are currently three doctors, two nurses, two chiropodists and two receptionists working at any given time. [2] [3]

The following are assumptions about Walkinstown Medical Centre. All documents and files are typed and printed and kept in a file folder on premises. This includes all schedules for patients, patient files, medication lists and vaccine information. All prescriptions are hand-written by a doctor. They are also using a standard phone system, so a lot of the time, a caller would receive an engaged tone.

This consumes a lot of time for both staff and patients which could be used for other services and assisting with patients on a faster basis. Walkinstown Medical Centre is currently not using any temperature devices for checking for COVID before entering the premises.

To keep all the services simplistic, AWS will mostly be used to demonstrate cloud-based services. Another cloud-based platform like Microsoft Aure, Google Cloud and IBM Cloud could all be used instead.

## Functional Requirements

### Requirement 1: Create appointment

#### Description & Priority

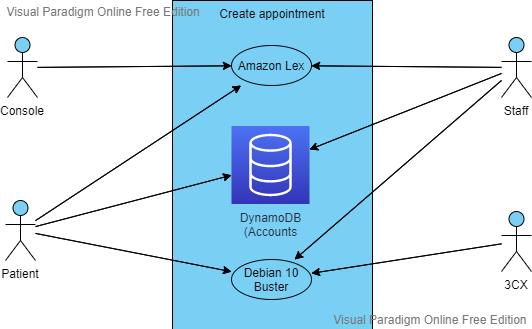
This requirement handles the appointments of patients who are visiting Walkinstown Medical Centre. This is the most important requirement as this would free up a lot of the staff’s time.

#### Use Case

**Unique ID:** create-appt

**Scope:** The scope of this requirement is to detail how a patient can create an appointment.

**Use Case Diagram:**

****

(Larger image in Appendix)

**Flow Description**

**Precondition**

* AWS is active and working correctly
* AWS’ relevant EC2 instances are active and running
* The patient’s device has access to Walkinstown Medical Centre’s website
* The Console (an Android tablet), where a patient check-in on premises, is connected to the Internet and Amazon Lex.

**Activation**

This use case begins when a patient navigates to book an appointment on Walkinstown Medical Centre’s website.

**Main flow**

1. The patient opens the website to the “Book an appointment” page. <See A1>
2. The patient selects the relevant service they require (GP, nurse, or chiropodist)
3. The patient selects a date and time they wish to book for.
4. The System checks with Amazon Lex to see if the appointment date and time are available
5. The System notifies the patient the date and time selected are available. <See A2>
6. The patient fills in their details
7. The System checks with DynamoDB if the patient already exists in the System. <See A3>
8. The System confirms the appointment and sends an email to the patient.

<Later at date and time of appointment>

1. Patient arrives at Walkinstown Medical Centre
2. Patient taps on Console to activate it.
3. Patient enters their date of birth and gender.
4. The Console checks with Amazon Lex if appointment exists <See A4>
5. The patient waits for service (doctor, nurse, or chiropodist)

**Alternative flow**

**<A1: Patient phones centre>**

1. The patient accesses the 3CX service by phoning Walkinstown Medical Centre.
2. The patient presses 1 to book an appointment.

<Returns to step 2 in Main flow, except voice commands prompt the patient>

**<A2: Selected date and time unavailable>**

1. The system notifies the patient the date and time are unavailable.

<Returns to step 3 in Main flow>

**<A3: Patient is not registered with centre>**

1. The System notifies the patient that they are not registered with the centre and requests the patient to phone to continue.

<Use Case ends>

**<A4: Console cannot find appointment>**

1. The Console notifies the patient the appointment cannot be found and to consult with reception

<Use Case ends>

**Termination**

This use case ends when a patient has successfully booked an appointment and is waiting to be seen by a service on-premises.

**Post condition**

This system goes into a wait state.

### Requirement 2: Phone system with 3CX

#### Description & Priority

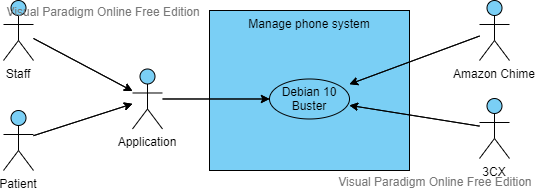
This requirement manages the phone system for Walkinstown Medical Centre.

#### Use Case

**Unique ID:** manage-phone

**Scope:** The scope of this requirement is to detail the phone system with Amazon Chime and 3CX

**Use Case Diagram:**

****

(Larger image in Appendix. “Application” refers to a computer or mobile application to use 3CX with, or a standard landline phone.)

**Flow Description**

**Precondition**

* AWS is active and working correctly
* AWS’ EC2 with Debian 10 Buster image is active and running.
* 3CX is active, working and set up correctly.
* The phone number for Walkinstown is ported to the Amazon Chime service and is active and working correctly.

**Activation**

This use case begins when a patient phones Walkinstown Medical Centre

**Main flow**

1. The patient phones Walkinstown Medical Centre.
2. The System greets them with an automated message.
3. The System prompts the user to select the service they require (GP, nurse, chiropodist) <See A1><See A2>
4. The patient selects an option
5. The System connected to the correct extension number. <See A3>

**Alternative flow**

**<A1: Patient phones centre>**

1. The System informs the patient that the option they have selected is not valid

<Returns to step 3 in Main flow>

**<A2: The centre is closed>**

1. The system notifies the patient the centre is closed and informs the patient of the opening times

<Use case ends>

**<A3: Extension number unanswered after 30 seconds>**

1. The System notifies the patient that the call was not answered and to try again later.

<Use Case ends>

**Termination**

This use case ends when a staff member answers the call.

**Post condition**

This system goes into a wait state.

### Requirement 3: Accounts

#### Description & Priority

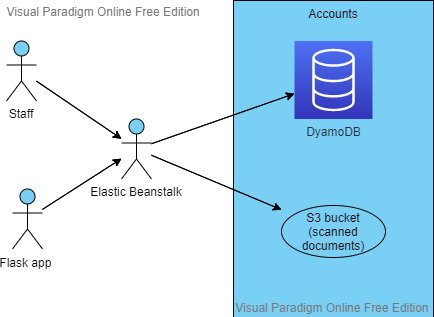
This requirement manages the accounts for Walkinstown Medical Centre.

#### Use Case

**Unique ID:** accounts

**Scope:** The scope of this requirement is to detail accessing accounts using DynamoDB and S3 in a Flask app deployed on Elastic Beanstalk

**Use Case Diagram:**

****

(Larger image in Appendix.)

**Flow Description**

**Precondition**

* The Flask app is active, running, and accessible.
* AWS for DynamoDB and S3 are active.

**Activation**

This use case begins when a staff member opens the Flask app on the Elastic Beanstalk website.

**Main flow**

1. The staff member enters their username and password.
2. The System logs the staff member in. <See A1>
3. The System displays navigation links for patient accounts, medications, vaccine records and uploaded documents
4. The staff member clicks on patient accounts
5. The System shown a search bar.
6. The staff member enters the surname of the patient and selects “Search”.
7. The System queries the DynamoDB and returns the relevant results. <See A2>
8. The staff member clicks on the requested patient’s name.
9. The system displays the patient account information.

**Alternative flow**

**<A1: Staff member’s username or password incorrect>**

1. The staff member clicks on “Forgot password”
2. The System displays an input page for the staff member’s username.
3. The staff member enters in their username.
4. The System sends an email with the current password to the staff member’s email address.

<Returns to step 1 in Main flow>

**<A2: No results for searched surname >**

1. The System displays to the staff member there are no results.

< Returns to step 6 in Main flow >

**Termination**

This use case ends when a staff member answers the call.

**Post condition**

This system goes into a wait state.

### Requirement 4: Amazon Rekognition

#### Description & Priority

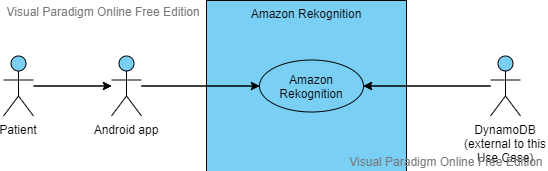
This requirement details how a patient enters the centre while using an Android tablet with an app using Amazon Rekognition to check they are wearing a face mask and to check their temperature.

#### Use Case

**Unique ID:** rekognition

**Scope:** The scope of this requirement is to detail how Amazon Rekognition is used on an Android app to check whether a patient is wearing a face mask and to check their temperature.

**Use Case Diagram:**

****

(Larger image in Appendix.)

**Flow Description**

**Precondition**

* The Flask app is active, running, and accessible.
* AWS for DynamoDB and S3 are active.
* AWS Rekognition is active and running.
* The Android tablet is on, connected to Wi-Fi, app open and relevant permissions for the app (e.g., camera, Internet) are allowed.

**Activation**

This use case begins when the app detects a patient’s face.

**Main flow**

1. The patient approaches the Android tablet.
2. The System detects whether the patient is wearing a face mask. <See A1>
3. The System generates a green box with a “Face mask” label around the mask.
4. The System detects the person’s temperature <See A2>
5. The System displays a light green text box with the patient’s temperature at the top of the screen.
6. The System waits 5 seconds to ensure a correct reading. <See A3>
7. The System displays a light green box with the text “Please enter”
8. The System sends the face mask information (“Present” or “None”) and temperature information to a DynamoDB.

**Alternative flow**

**<A1: Patient not wearing mask>**

1. The System generates a red box with a “Face mask” label around the mouth area.
2. The System displays a light red box with the text “You must wear a face mask to enter”.

<Use Case ends>

**<A2: Patient’s temperature is too high.>**

1. The System displays a light red text box with the patient’s temperature at the top of the screen.
2. The System displays a light red box with the text “Your temperature is too high, please go home and phone the centre instead of entering”.

<Use case ends>

**Termination**

This use case ends when the patient enters the centre.

**Post condition**

This system goes into a wait state.

### Requirement 5: VPN

#### Description & Priority

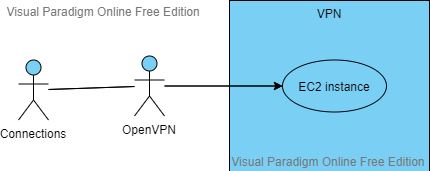
This requirement details the VPN using and EC2 instance and OpenVPN.

#### Use Case

**Unique ID:** vpn

**Scope:** The scope of this use case is to detail the VPN.

**Use Case Diagram:**

****

(Larger image in Appendix.)

**Flow Description**

**Precondition**

* The EC2 instance is active and running
* OpenVPN on the EC2 instance is active and running.

**Activation**

This use case begins when a staff member opens the Flask app on the Elastic Beanstalk website.

**Main flow**

1. The VPN is active. <See A1>

**Alternative flow**

**<A1: VPN is not active>**

1. A staff member checks if the VPN is active or is currently having issues and resolves the issue.

<Returns to step 1 in Main flow>

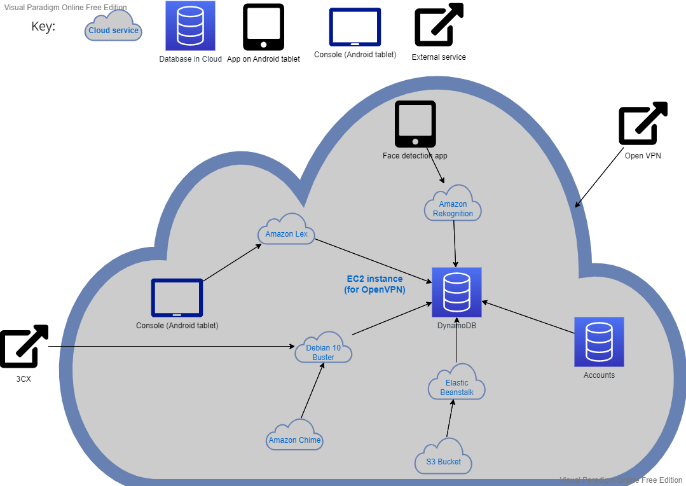
**Termination**

This use case ends when the VPN is active.

**Post condition**

The VPN is active.

# Architetectural design

  
(Larger image in Appendix.)

To move to a cloud-based system, Walkinstown Medical Centre should consider these services:

## Amazon Lex

Amazon Lex is a service that allows a developer to create applications that use text or voice commands to act on behalf of a user. A bot does these actions (intents). It uses the same technology that drives Amazon Alexa. An example of this is where a bot could order a pizza. Parameters or “slots” can also be added such as a date and time, thin pizza, add pepperoni, no cheese, and so on. [4]

Amazon Lex will be used to create the appointment system for Walkinstown Medical Centre.

### Advantages

* Advantages of Amazon Lex include being able to build smart devices apps, including incorporating Lex into iOS, Android, Java, Python, .NET, PHP, C++ creating applications for a while array of computer systems. [5]
* Lex supports techniques to analyse natural language and speech [5]

### Disadvantages

* Does not support other languages and is only in English [5]
* Web integration is difficult to comprehend [5]

## Debian 10 Buster with 3CX and Amazon Chime

3CX is a phone system that can be installed on a Linux machine, in this case a Debian machine. With 3CX, one can have many extension numbers for their business, as well as automated caller prompts, voicemail, out-of-office greetings, SMS, live chat, and video calls, to name a few features. It also provides a variety of related services. 3CX can be used with a smartphone, PC, or a traditional landline. [6]

If one is using AWS, to get started with 3CX, one will need a Debian 10 Buster license from the AWS Marketplace. After that, it's a simple matter of setting up 3CX as 3CX does most of the work. There are tutorials available on the internet. [7]

Amazon Chime is a secure all-in-one communications service that lets you meet, chat, and conduct business calls both inside and outside of a company. Chime works on multiple platforms allowing a company to stay connected even when not on-site. One can “provision” or order a phone number or port their current number. [8]

For Walkinstown Medical Centre, a porting of their current number to be used with Amazon Chime would be most beneficial as the centre has had this number for many years.

### Advantages of Amazon Chime

* Chime can be used with any operating system [9]
* Supports recording of video calls and allows one to share their screen. [9]
* Has the ability to use remote desktop functions. [9]
* Supports the use of chat rooms [9]

### Disadvantages of Amazon Chime

* The API is private meaning one must use it with AWS. [9]
* Only English is supported [9]
* Lacking ability to use a drawing feature like a whiteboard. [9]

## Amazon DynamoDB and S3

Amazon DynamoDB is a NoSQL database service that can be used with AWS. DynamoDB is named after Dynamo and features a data concept that is like Dynamo, but it is implemented differently. [10]

S3 is a scalable storage infrastructure from AWS. Amazon.com's global e-commerce network runs on the same infrastructure. S3 can store any sort of item, making it ideal for Internet application storage, backup and recovery, disaster recovery, data archives, analytics data lakes, and hybrid cloud storage. S3 Buckets are a resource that contains objects. [11]

DynmaoDB and S3 will be used to manage the likes of patient accounts, medical lists, vaccine records, scanned and uploaded documents as well as store a copy of prescriptions given to patients.

AWS Elastic Beanstalk is an Amazon Web Services utility for automating the configuration, coordination, and administration of computer systems and software. It controls several AWS services, including management of EC2 instances, adding, or removing S3 objects, automatic scaling, Simple Notification Service (SNS), Elastic Load Balancers to name a few. [12]

A web application will be created for Walkinstown Medical Centre using the micro-web framework Flask. The staff will be able to access the accounts via this Flask app after it has been deployed to Elastic Beanstalk.

### Advantages of DynamoDB

* DynamoDB is extremely easy to set up [13]
* With Using DynamoDB, hosting expenses for small organisations may be drastically lowered. [13]
* DynamoDB supports a reaction to when another system’s data changes, keeping both uniform. [13]

### Disadvantages of DynamoDB

* When compared to a classic SQL database or MongoDB, for example, creating an ad-hoc querying system using DynamoDB is practically impossible. [13]
* As one is “paying per usage”, the actual cost of using DynamoDB is difficult to forecast. Even though AWS has a “free-tier” option, fixing errors and testing can result in going beyond the free-tier limits. [13]

## Amazon Rekognition

Amazon Rekognition is a computer vision system that was released in 2016 as a cloud-based SaaS. It is used to recognise faces and their facial expressions, check if one is wearing personal protective equipment, add labels, and many more features. Many government agencies in the USA use Rekognition, such as US Immigration and Customs Enforcement and Orlando police in Florida. [14] [15]

An Android tablet with an app will be using Amazon Rekognition to detect a person’s face and to gauge their temperature. [16] It will be placed in a custom-made stand in the porch area of the centre. Future developments could open the main door only when the temperature reading is within a certain limit. This would prove beneficial during the COVID pandemic.

### Advantages of Amazon Rekognition

* Rekognition can look for faces, celebrities, text, objects, PPE equipment among others. [17]

### Disadvantages of Amazon Rekognition

* Sometimes Rekognition does not always get it right as sometimes it identified white men as black women. [17]

## Amazon EC2 instance with a OpenVPN service

OpenVPN is a VPN system that includes ways for creating secure point-to-point or site-to-site connections in routed or bridged configurations, as well as remote access. It can execute both client and server applications. [18]

One can set up OpenVPN with an EC2 instance by going to the Amazon Marketplace and following a tutorial, which can be found online. [19]

### Advantages of OpenVPN

* Security with OpenVPN is paramount. Open VPN uses “256-bit encryption keys”. This means that it is difficult to for the likes of a hacker to interpret a connection [20]
* OpenVPN is capable of being used of varies operating systems including MacOS, iOS and Android. [20]

### Disadvantages of OpenVPN

* Unfortunately, OpenVPN does not come as a stand-alone program and put be inserted into an operating system. [20]
* To create custom settings with OpenVPN can be troublesome. OpenVPN is abundant with options and configuration settings that the “average Joe” may not be able to complete. [20]
* OpenVPN limits the amount of services for free customers and can only handle a maximum of 50 servers for per licence. [20]

# Conclusion

We looked at how a small medical centre may automate their manual processes and move them to cloud-based alternatives. We investigated how Amazon Lex could be used to manage Walkinstown Medical Centre's appointment system, as well as how 3CX and Amazon Chime could be used to migrate the centre’s current basic phone system to the cloud, allowing patients to call and be correctly redirected even when the receptionists are busy.

We also looked at how DynamoDB and S3 could be used to store patient accounts, vaccination records, and other data in the cloud. With Amazon Rekognition, an Android tablet could recognise if a patient was wearing a face mask and if their temperature was normal. We also talked about how an EC2 coupled to an OpenVPN may provide an extra layer of internet security to the centre.

It was learned that AWS is not perfect and that some other cloud providers may do a better job. Would this author recommend AWS to Walkinstown Medical Centre? Honestly, probably not. It would be more beneficial to look at all cloud providers and see which one would fit the centre’s needs, instead of trying to push AWS. It was enjoyed how Amazon Lex and Alexa are similar and how they are in totally different markets.

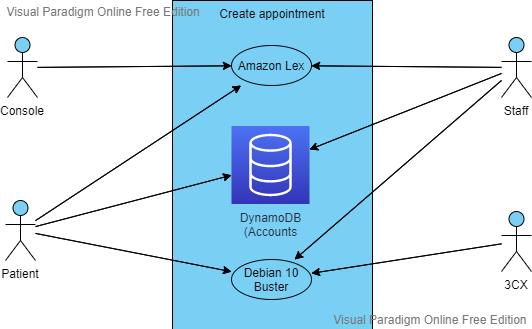
The author tested the EC2 instance with OpenVPN. A problem arose when just generally browsing the Internet with the VPN active. Many websites mistook the OpenVPN hosted on AWS as spam. A substantial amount of extra verification to prove the author was not a bot occurred. It was not practical for the author’s needs, but a good learning experience.

The author also tested the 3CX with Amazon Chime. Honestly, it’s faultless. Highly recommend. The author now has a New York phone number and a business WhatsApp account. Very easy and doable. The only snag that occurred was provisioning the phone number. New AWS accounts are not permitted to do this as a restriction is put onto the account. One needs to contact Amazon to remove this restriction.

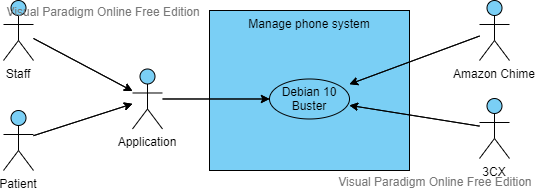
# References

|  |  |
| --- | --- |
| [1] | M. Thomas, “Six Ways Cloud Computing is Improving Healthcare,” Built In, 8 April 2020. [Online]. Available: https://builtin.com/cloud-computing/cloud-computing-in-healthcare. [Accessed 10 January 2022]. |
| [2] | Golden Pages, “Gough, Mary - BSc Pod Med,” Golden Pages, 9 January 2022. [Online]. Available: https://www.goldenpages.ie/gough-mary-bsc-pod-med-dublin-D12/. [Accessed 9 January 2022]. |
| [3] | Google, “Walkinstown Medical Centre - Reviews,” Google, 9 January 2022. [Online]. Available: https://www.google.com/search?q=walkinstown+medical+centre. [Accessed 9 January 2022]. |
| [4] | Amazon, “Amazon Lex: How It Works,” Amazon, 9 January 2022. [Online]. Available: https://docs.aws.amazon.com/lex/latest/dg/how-it-works.html. [Accessed 9 January 2022]. |
| [5] | Intellectyx, “AMAZON LEX,” Intellectyx, 10 January 2022. [Online]. Available: https://www.intellectyx.com/techstack/amazon-lex/. [Accessed 10 January 2022]. |
| [6] | 3CX, “PBX Reinvented: More Than Voice,” 3CX, 9 January 2022. [Online]. Available: https://www.3cx.com/pbx/. [Accessed 9 January 2022]. |
| [7] | 3CX, “Hosting 3CX in Amazon AWS,” 3CX, 9 January 2022. [Online]. Available: https://www.3cx.com/docs/cloud-pbx-amazon-aws/. [Accessed 9 January 2022]. |
| [8] | Amazon, “What is Amazon Chime?,” Amazon, 10 January 2022. [Online]. Available: https://docs.aws.amazon.com/chime/latest/ug/what-is-chime.html. [Accessed 10 January 2022]. |
| [9] | J. Martinez, “Amazon Chime Review,” PCMag, 14 March 2017. [Online]. Available: https://www.pcmag.com/reviews/amazon-chime. [Accessed 9 January 2022]. |
| [10] | Amazon, “What Is Amazon DynamoDB?,” Amazon, 9 January 2022. [Online]. Available: https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html. [Accessed 9 January 2022]. |
| [11] | Amazon, “What is Amazon S3?,” Amazon, 9 January 2022. [Online]. Available: https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html. [Accessed 9 January 2022]. |
| [12] | Amazon, “What is AWS Elastic Beanstalk?,” Amazon, 9 January 2022. [Online]. Available: https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/Welcome.html. [Accessed 9 January 2022]. |
| [13] | M. Bonig, “The Pros and Cons of DynamoDB,” Matthew Bonig, 28 June 2019. [Online]. Available: https://matthewbonig.com/2019/06/28/dynamodb-pros-cons/. [Accessed 10 January 2022]. |
| [14] | F. Lardinois, “Amazon launches Amazon AI to bring its machine learning smarts to developers,” TechCrunch, 30 November 2016. [Online]. Available: https://techcrunch.com/2016/11/30/amazon-launches-amazon-ai-to-bring-its-machine-learning-smarts-to-developers/. [Accessed 9 January 2022]. |
| [15] | Amazon, “What is Amazon Rekognition?,” Amazon, 9 January 2022. [Online]. Available: https://docs.aws.amazon.com/rekognition/latest/dg/what-is.html. [Accessed 9 January 2022]. |
| [16] | A. SA, “AWS Rekognition in Android,” Medium, 25 May 2019. [Online]. Available: https://medium.com/how-to-integrate-aws-rekognition-in-android/aws-rekognition-in-android-9d16f16d591c. [Accessed 9 January 2022]. |
| [17] | S. Patil, “What is AWS Rekognition, Pros and Cons and Why it is banned ?,” Saurav's Online, 11 June 2020. [Online]. Available: https://sauravsonline.wordpress.com/2020/06/11/what-is-aws-rekognition-pros-and-cons-and-why-it-is-banned/. [Accessed 10 January 2022]. |
| [18] | OpenVPN, “We’re building a more secure internet,” OpenVPN, 9 January 2022. [Online]. Available: https://openvpn.net/about/. [Accessed 9 January 2022]. |
| [19] | OpenVPN, “Launch OpenVPN Access Server BYOL on Amazon AWS,” OpenVPN, 9 January 2022. [Online]. Available: https://openvpn.net/aws-video-tutorials/byol/. [Accessed 9 January 2022]. |
| [20] | S. George, “OpenVPN – Pros and Cons,” Bob Cares, 8 February 2019. [Online]. Available: https://bobcares.com/blog/openvpn-pros-and-cons/. [Accessed 9 January 2022]. |

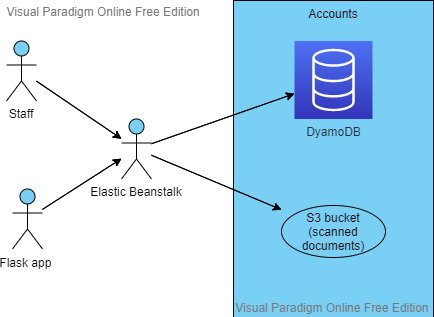
# Appendix



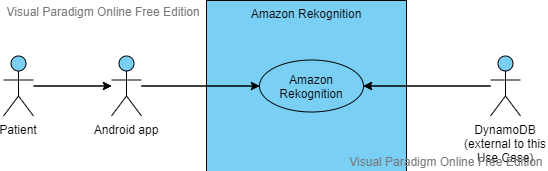
Requirement 1: Create appointment Use Case Diagram



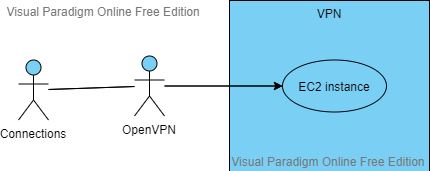
Requirement 2: Phone system with 3CX Use Case Diagram



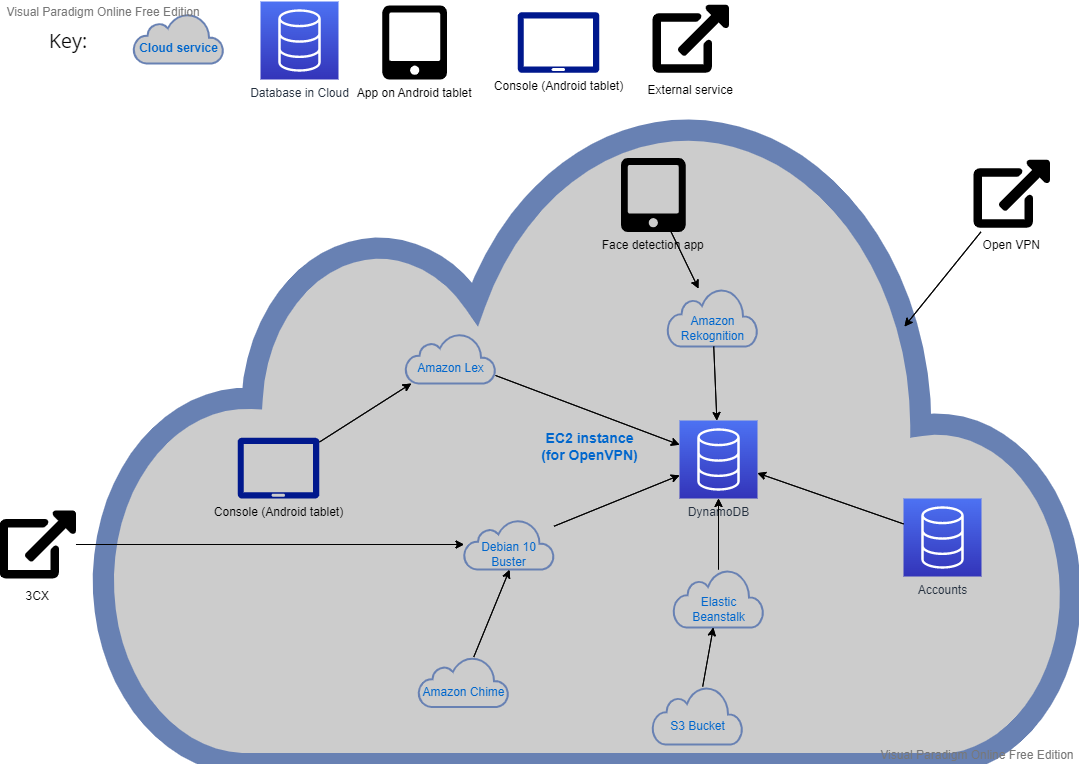
Accounts Use Case Diagram



Amazon Rekognition Use Case Diagram



VPN Use Case Diagram



Architectural Design.