# Disease Screening

## Contents

# 1. Introduction

## 1.1. Business Scenario

A patient with an unknown disease schedules an appointment in a hospital or medical institution.

The doctor attends the patient and evaluates his/her symptoms in order to discern if they are compatible with a particular disease or another one with any or all similar external signs, and makes a first diagnosis.

Then, the doctor acts accordingly, for example by prescribing a medicine, scheduling some tests to confirm the first diagnosis, an appointment with a specialist, or recommending some rest period.

## 1.2. Opportunity

During the **Covid-19 Pandemic**, a medical institution needs to **quickly** evaluate if a patient has symptoms compatible with that particular disease or another one with any similar external signs (flu, cold, allergy, asthma, anxiety, etc.).

In addition, the hospital needs to do it **remotely**, without jeopardizing the health of its employees or the patient, by using channels such us the telephone, a mobile application or a web form.

Then, the medical institution should acts accordingly, for example, by scheduling some tests to confirm the first diagnosis, an appointment with the doctor or a specialist, or recommending some period of confinement.

## 1.3. Solution Approach

A **quick and secure** way of doing the disease screening based on the symptoms, without the needed of came face to face with the patient, is to use **Big Data** and a decisioning **Adaptive Model**, and make the screening remotely through **NLP** (Natural Language Processing) in a **chat bot** hosted in the hospital website. We will create an application with a new Disease Screening case type in order to manage it.

As an additional screening aid service, the patient could query how many known positive cases of Covid-19 are or were in a given location and date where he/she could have been in contact with. The results will be based on the confirmed cases in our application, plus data imported from the hospital database, if it is possible.

As a result of the first diagnosis, the solution should be able of scheduling an appointment in the medical institution system, and retrieve the final diagnosis in order to improve the reliability of the adaptive model and the Covid-19 positives localization tool.

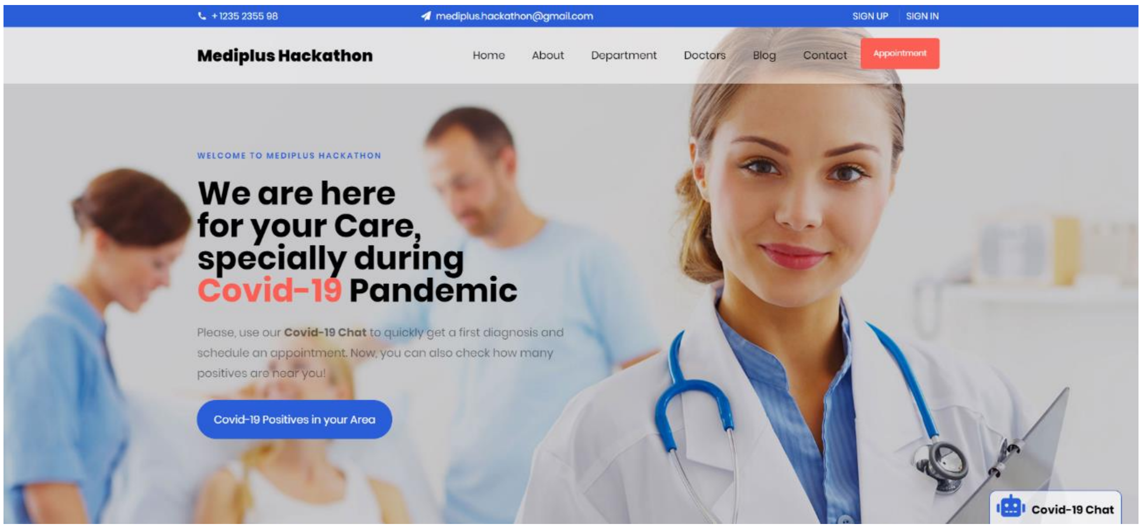
## 1.4. Benefits

- **Security**: the first diagnosis is made remotely, without jeopardizing the health of the hospital employees or the patient.
- **Speed**: it is not necessary to schedule an appointment and visit the doctor, the first diagnosis can be made by the patient in a few minutes.
- **Reliability**: the use of Big Data and an adaptive model provides accurate results (the feedback from the hospital helps in improving the predictions).

- **Privacy**: sensitive patient information is never shared with third parties.
- **Accessibility**: the diagnosis can be done anywhere and anytime from a mobile phone, a tablet or a laptop.
- **Scalability**: the solution is open enough to increase the number of recognized diseases, and connect with any medical institution managing system.

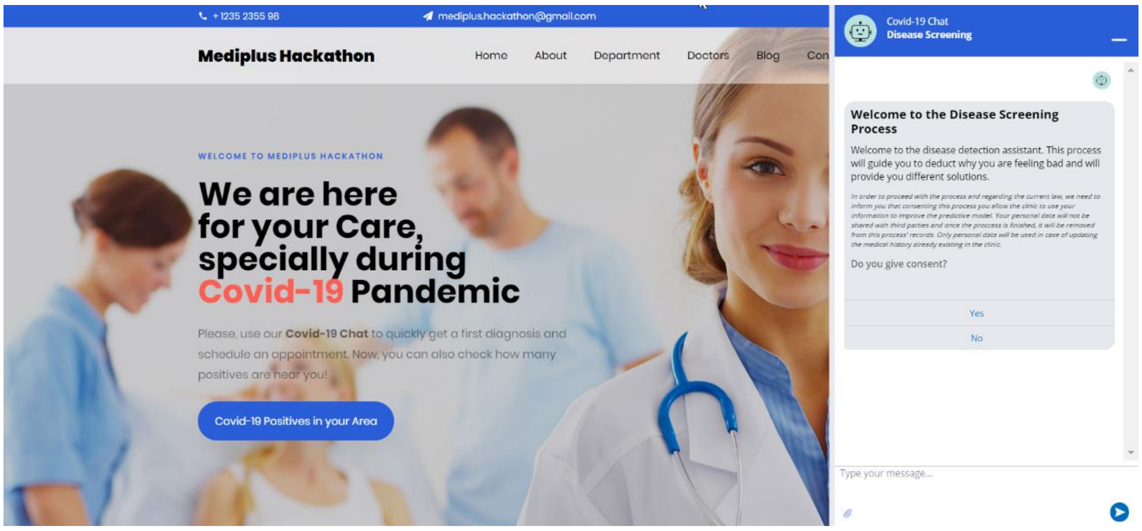
## 2. Functional Behavior. Case of Use

One person, the patient from now on, access the website of the medical institution. There, the patient will find our chat bot available to be open.


+1235 2355 98
mediplus.hackathon@gmail.com
SIGN UP
SIGN IN
Mediplus Hackathon
Home
About
Department
Doctors
Blog
Contact
Appointment
WELCOME TO MEDIPLUS HACKATHON
We are here for your Care, specially during Covid-19 Pandemic
Covid-19 Positives in your Area
Covid-19 Chat

Please, use our **Covid-19 Chat** to quickly get a first diagnosis and schedule an appointment. Now, you can also check how many positives are near you!

When clicking on it, the chat bot will present itself, indicating its function and asking for permission to treat the personal data entered by the patient. This data will be only used for our process and the hospital and will not be shared with third parties.


+1235 2355 98
mediplus.hackathon@gmail.com
Mediplus Hackathon
Home
About
Department
Doctors
Blog
Con
WELCOME TO MEDIPLUS HACKATHON
We are here for your Care, specially during Covid-19 Pandemic
Covid-19 Positives in your Area
Covid-19 Chat
Disease Screening
Yes
No
Type your message...

Please, use our **Covid-19 Chat** to quickly get a first diagnosis and schedule an appointment. Now, you can also check how many positives are near you!

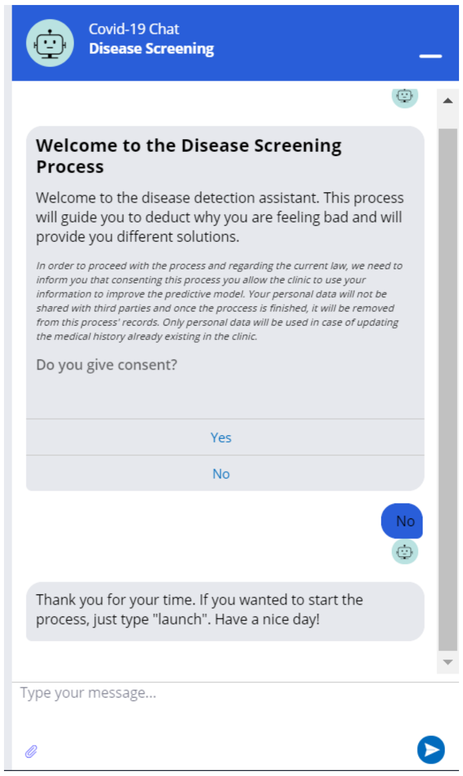
**Welcome to the Disease Screening Process**

Welcome to the disease detection assistant. This process will guide you to deduct why you are feeling bad and will provide you different solutions.

*In order to proceed with the process and regarding the current law, we need to inform you that consenting this process you allow the clinic to use your information to improve the predictive model. Your personal data will not be shared with third parties and once the proccess is finished, it will be removed from this process' records. Only personal data will be used in case of updating the medical history already existing in the clinic.*

Do you give consent?

If the consent is not given, the chat will indicate that the process has finished and if the patient wanted to start it again, he/she should type "launch". When doing that, the same consent is asked.


Covid-19 Chat
Disease Screening

**Welcome to the Disease Screening Process**

Welcome to the disease detection assistant. This process will guide you to deduct why you are feeling bad and will provide you different solutions.

*In order to proceed with the process and regarding the current law, we need to inform you that consenting this process you allow the clinic to use your information to improve the predictive model. Your personal data will not be shared with third parties and once the proccess is finished, it will be removed from this process' records. Only personal data will be used in case of updating the medical history already existing in the clinic.*

Do you give consent?

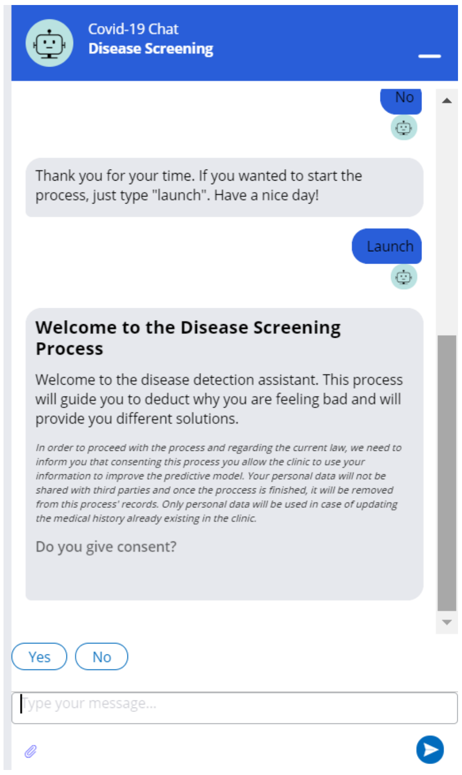
Yes

No

No

Thank you for your time. If you wanted to start the process, just type "launch". Have a nice day!

Type your message...


Covid-19 Chat
Disease Screening

No

Thank you for your time. If you wanted to start the process, just type "launch". Have a nice day!

Launch

**Welcome to the Disease Screening Process**

Welcome to the disease detection assistant. This process will guide you to deduct why you are feeling bad and will provide you different solutions.

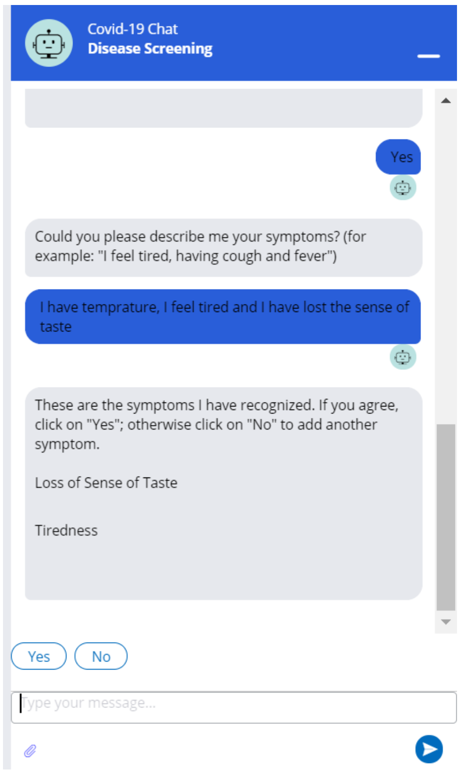
*In order to proceed with the process and regarding the current law, we need to inform you that consenting this process you allow the clinic to use your information to improve the predictive model. Your personal data will not be shared with third parties and once the proccess is finished, it will be removed from this process' records. Only personal data will be used in case of updating the medical history already existing in the clinic.*

Do you give consent?

Yes No

Type your message...

Once the patient gives the consent, the chat bot asks for the symptoms the patient is feeling. This one should introduce one sentence including the symptoms as the example given by the chat bot. For this example, you can insert: "I have *temprature*, I feel tired and I have lost the sense of taste". Please, typo "*temprature*" and not "temperature". The chat bot will show a list of recognized symptoms and the patient will be able to confirm them or enter again the symptoms.


Covid-19 Chat
Disease Screening

Yes

Could you please describe me your symptoms? (for example: "I feel tired, having cough and fever")

I have temprature, I feel tired and I have lost the sense of taste

These are the symptoms I have recognized. If you agree, click on "Yes"; otherwise click on "No" to add another symptom.

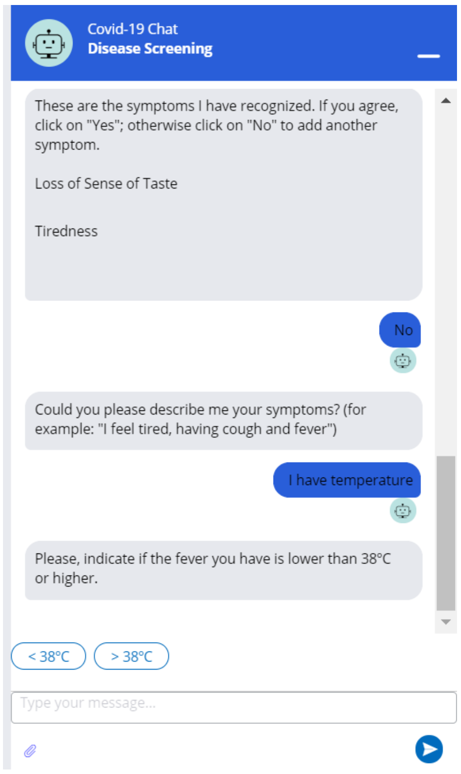
Loss of Sense of Taste

Tiredness

Yes No

Type your message...

In our case, we see that temperature was not recognized, so you can type "I have temperature" to add it. Then, the chat bot will give you options depending on the symptom that will affect the final diagnosis.



Covid-19 Chat
**Disease Screening**

These are the symptoms I have recognized. If you agree, click on "Yes"; otherwise click on "No" to add another symptom.

Loss of Sense of Taste

Tiredness

No

Could you please describe me your symptoms? (for example: "I feel tired, having cough and fever")

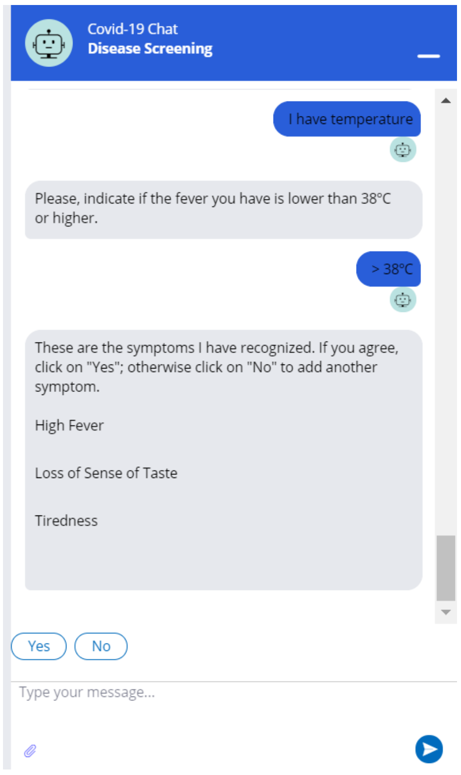
I have temperature

Please, indicate if the fever you have is lower than 38°C or higher.

< 38°C > 38°C

Type your message...

In this case, it is not the same having low fever than high fever, that is why, the chat bot asks about the fever. Again, the chat bot will show a list of the symptoms detected and the patient will be able to confirm them or not. If the chat bot still does not show all the symptoms introduced by the patient, it will ask again by them until the patient confirms them.



Covid-19 Chat
**Disease Screening**

I have temperature

Please, indicate if the fever you have is lower than 38°C or higher.

> 38°C

These are the symptoms I have recognized. If you agree, click on "Yes"; otherwise click on "No" to add another symptom.

High Fever

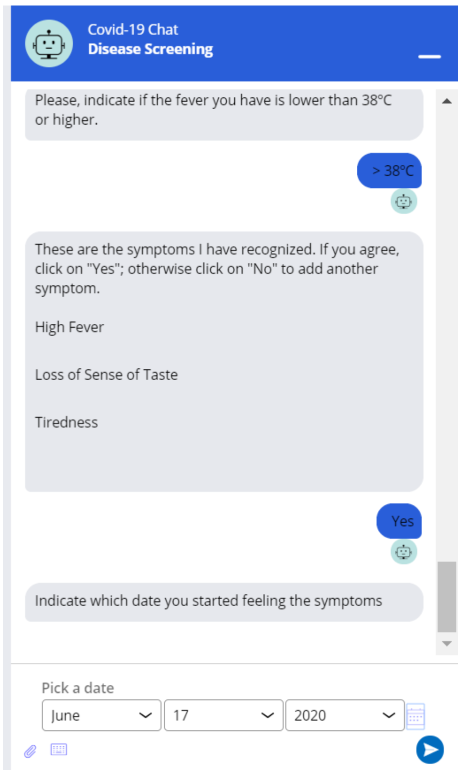
Loss of Sense of Taste

Tiredness

Yes No

Type your message...

Once confirmed, other information is needed before the diagnosis. Some basic information is the date when the symptoms appeared, the age, the gender and if the patient has been in contact with anyone diagnosed with Covid-19. This information must have the proper format or the chat bot will ask again for it until the format used is proper. In addition to all those questions, a list of countries is shown as "areas of risk". This list depends on the current location (the country) of the patient and must indicate if he/she has been in any of them recently.


Covid-19 Chat
**Disease Screening**

Please, indicate if the fever you have is lower than 38°C or higher.

> 38°C

These are the symptoms I have recognized. If you agree, click on "Yes"; otherwise click on "No" to add another symptom.

High Fever

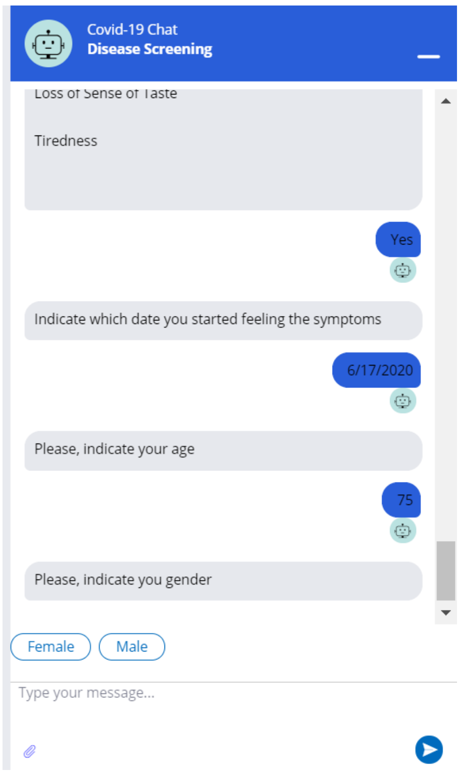
Loss of Sense of Taste

Tiredness

Yes

Indicate which date you started feeling the symptoms

Pick a date
June 17 2020


Covid-19 Chat
**Disease Screening**

Loss of Sense of Taste

Tiredness

Yes

Indicate which date you started feeling the symptoms

6/17/2020

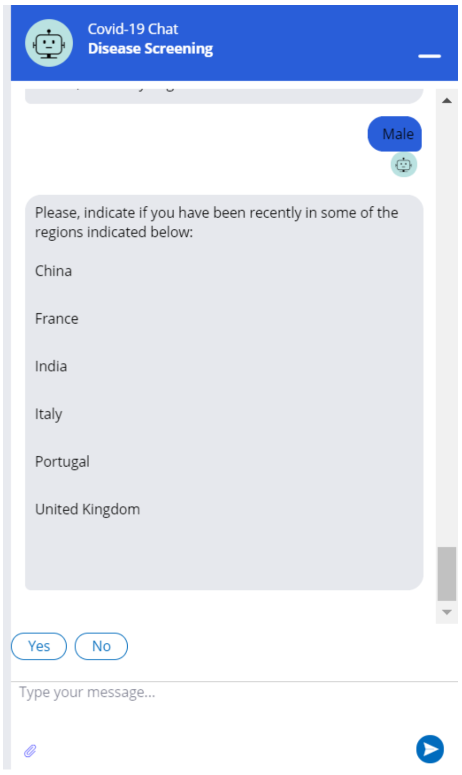
Please, indicate your age

75

Please, indicate you gender

Female Male

Type your message...


Covid-19 Chat
**Disease Screening**

Male

Please, indicate if you have been recently in some of the regions indicated below:

China

France

India

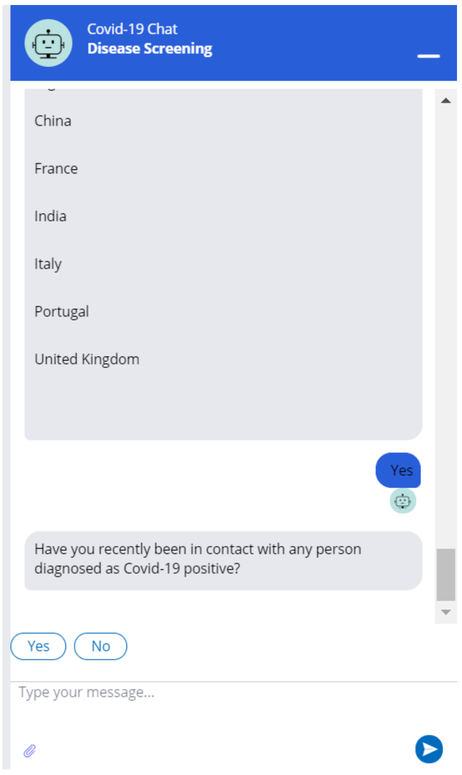
Italy

Portugal

United Kingdom

Yes No

Type your message...


Covid-19 Chat
**Disease Screening**

China

France

India

Italy

Portugal

United Kingdom

Yes

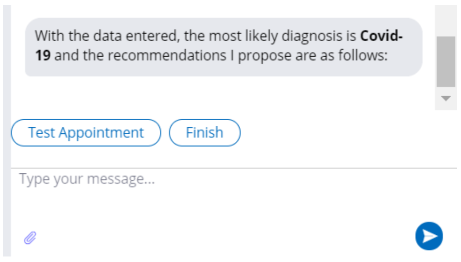
Have you recently been in contact with any person diagnosed as Covid-19 positive?

Yes No

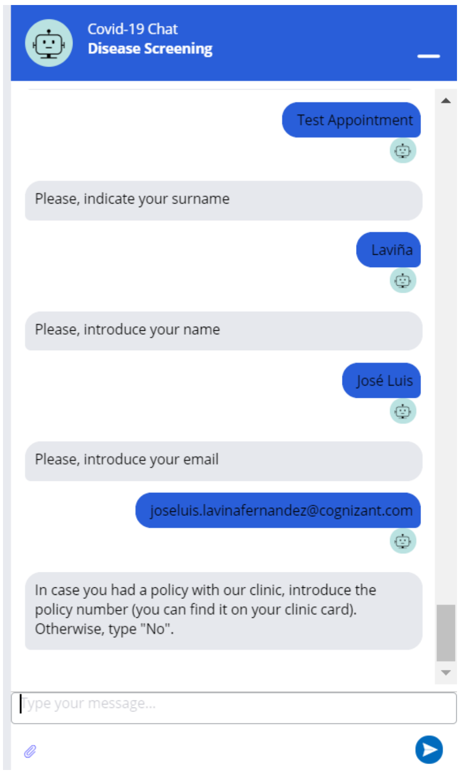
Type your message...

When the patient answers all of these questions, the chat bot will show the initial diagnosis (returned by the adaptive model) and will provide the patient some options (appointments) to choose, including the “finish” option in case the patient doesn’t want to proceed with the process and attend one appointment. As example, if the initial diagnosis is “Covid-19”, the option given is “Test Appointment”; if the diagnosis were “Allergy”, the options would be “Test Appointment” and “Specialist Appointment”. The different appointments considered are:

- General Appointment
- Specialist Appointment
- Test Appointment


With the data entered, the most likely diagnosis is **Covid-19** and the recommendations I propose are as follows:

As already said, if the patient chose “Finish” the case would not proceed. However, selecting any of the other options, the chat bot asks for additional information: the surname, name and policy number (in case the patient does not have a medical cover he could indicate it typing “no” on that question).



Test Appointment

Please, indicate your surname

Laviña

Please, introduce your name

José Luis

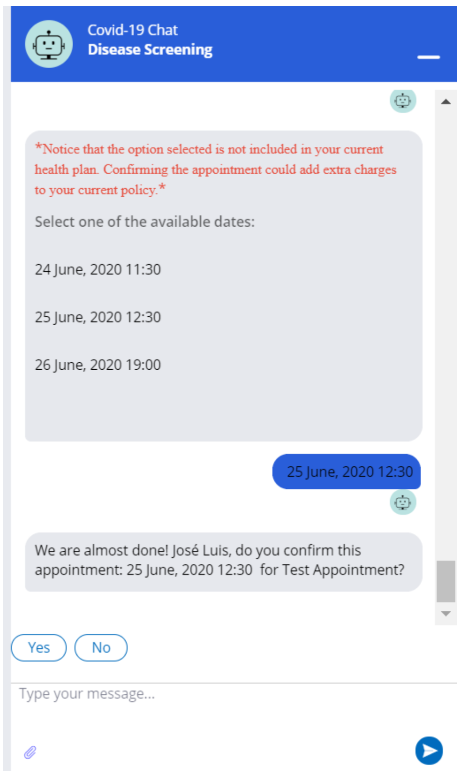
Please, introduce your email

joseluis.lavinafernandez@cognizant.com

In case you had a policy with our clinic, introduce the policy number (you can find it on your clinic card). Otherwise, type "No".

Then, the health plan information (coverage and guarantees) will be retrieved from the medical institution as so as the available dates for the type of appointment chosen. As these processes depend on each clinic, we have added several placeholder steps, and the real solution must be implemented in the future as part of the customization and integration phases. Also, a retry mechanism has not been implemented yet due to the diversity of scenarios and complexity of them.

In next step, once all the hospital information is retrieved, the chat bot will show to the patient some dates and times available for the appointment that the patient may choose. In addition, a red message comes indicating that the current health plan does not cover totally or partially the cost of that option if that is the case. If the patient is not sure about the option, he could reject that option in next question and select a different appointment.



*Notice that the option selected is not included in your current health plan. Confirming the appointment could add extra charges to your current policy.*

Select one of the available dates:

24 June, 2020 11:30

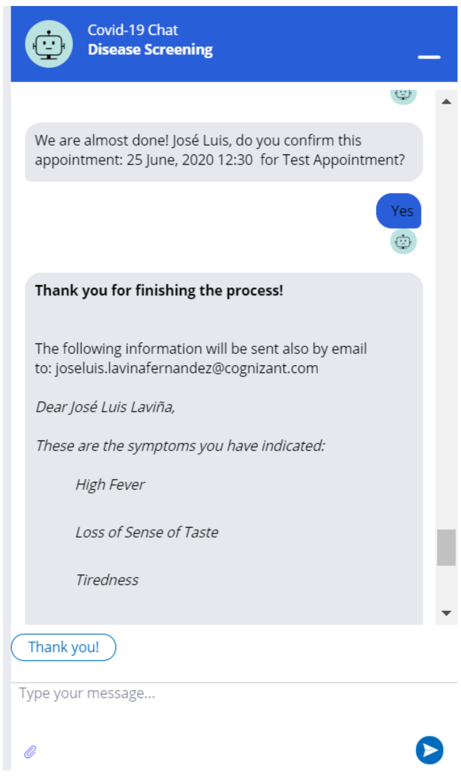
25 June, 2020 12:30

26 June, 2020 19:00

25 June, 2020 12:30

We are almost done! José Luis, do you confirm this appointment: 25 June, 2020 12:30 for Test Appointment?

When confirmed, the last screen shown is a summary of the case that will be sent also to the email provided previously on the chat.



We are almost done! José Luis, do you confirm this appointment: 25 June, 2020 12:30 for Test Appointment?

Yes

**Thank you for finishing the process!**

The following information will be sent also by email to: joseluis.lavinafernandez@cognizant.com

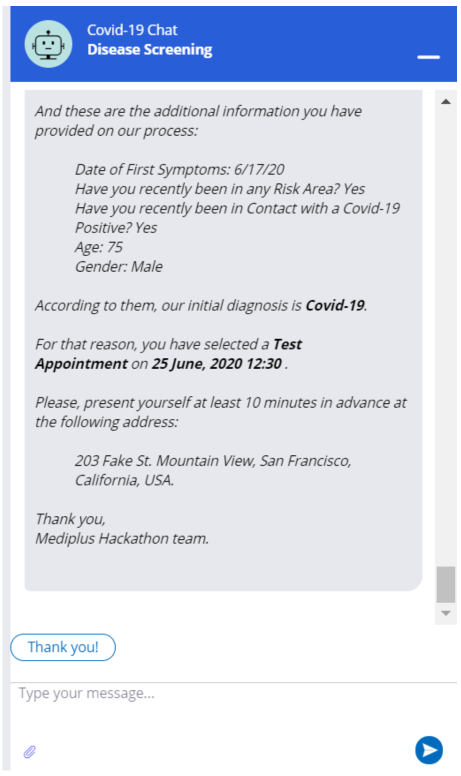
*Dear José Luis Laviña,*

*These are the symptoms you have indicated:*

*High Fever*

*Loss of Sense of Taste*

*Tiredness*



*And these are the additional information you have provided on our process:*

*Date of First Symptoms: 6/17/20*
*Have you recently been in any Risk Area? Yes*
*Have you recently been in Contact with a Covid-19 Positive? Yes*
*Age: 75*
*Gender: Male*

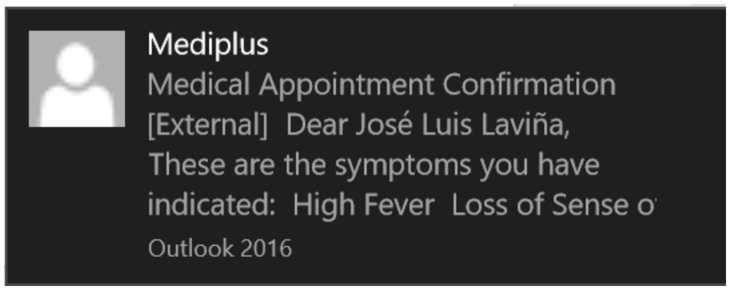
*According to them, our initial diagnosis is **Covid-19**.*

*For that reason, you have selected a **Test Appointment** on **25 June, 2020 12:30**.*

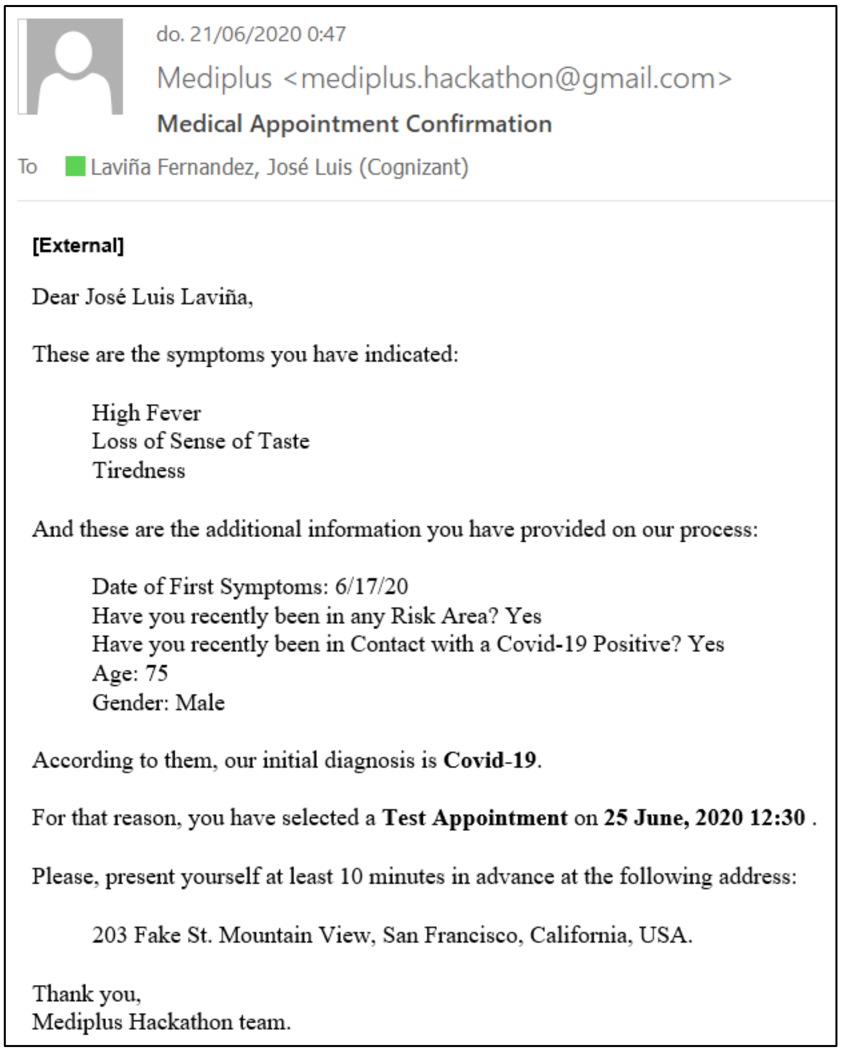
*Please, present yourself at least 10 minutes in advance at the following address:*

*203 Fake St. Mountain View, San Francisco, California, USA.*

*Thank you,*
*Mediplus Hackathon team.*



Mediplus
Medical Appointment Confirmation
[External] Dear José Luis Laviña,
These are the symptoms you have
indicated: High Fever Loss of Sense o
Outlook 2016



do. 21/06/2020 0:47
Mediplus <mediplus.hackathon@gmail.com>
**Medical Appointment Confirmation**
To 🟩 Laviña Fernandez, José Luis (Cognizant)

**[External]**

Dear José Luis Laviña,

These are the symptoms you have indicated:

High Fever
Loss of Sense of Taste
Tiredness

And these are the additional information you have provided on our process:

Date of First Symptoms: 6/17/20
Have you recently been in any Risk Area? Yes
Have you recently been in Contact with a Covid-19 Positive? Yes
Age: 75
Gender: Male

According to them, our initial diagnosis is **Covid-19**.

For that reason, you have selected a **Test Appointment** on **25 June, 2020 12:30** .

Please, present yourself at least 10 minutes in advance at the following address:

203 Fake St. Mountain View, San Francisco, California, USA.

Thank you,
Mediplus Hackathon team.

You can start a new process typing “launch”.

## 3. Technical Explanation. Implementation.

The idea of this section is to give a general description of how we have implemented this solution.

### 3.1. Data Model

We have created several data classes to store and manage all the necessary information:

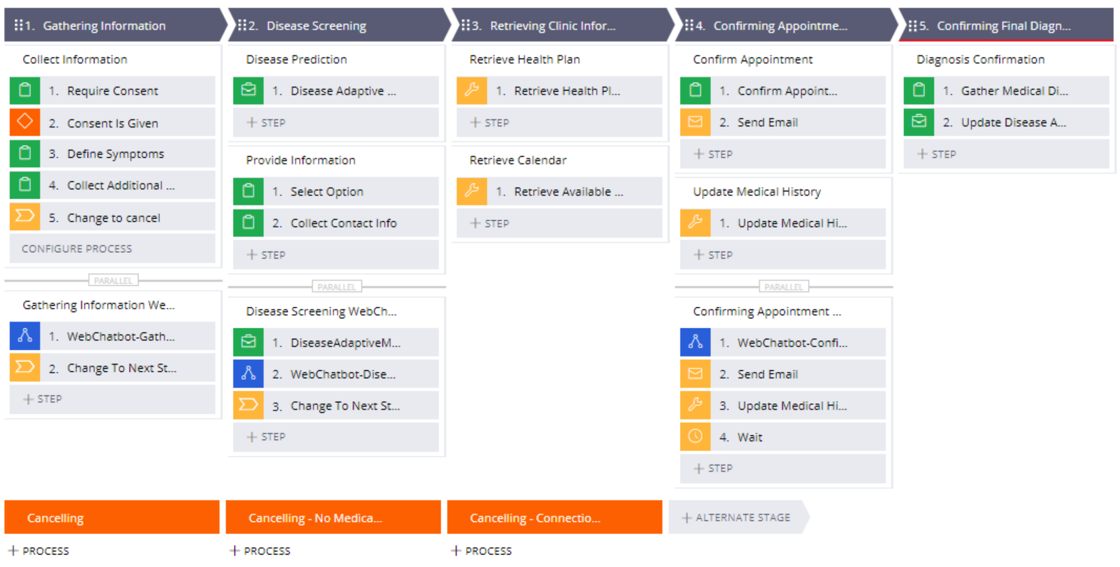
- **Pacient** (based on **CTS-Covid19DA-Data-DiagnosisRecord**). It will record most of the case information including the initial symptoms given by the patient, personal information, first diagnosis (given by the adaptive model) and option selected.
- **Symptoms** (based on **CTS-Covid19DA-Data-Symptoms**). It includes all the symptoms that may be recognized by the chat as True/False properties, and used by the adaptive model later (see Appendix A: Symptoms).
- **ClinicInformation** (based on **CTS-Covid19DA-Data-ClinicInformation**). This property will store all the information associated to the medical institution and to the patient's health plan (in case the patient had one).

In addition to the data classes, we have created two different data types:

- **Areas of Risk**: Used to store the risk countries depending on the origin country.
- **Disease**: Here we store the list of diseases used by the adaptive model (see Appendix B: Diseases).

## 3.2. Case Type. Disease Screening

The **Disease Screening** case type is built in the **CTS-Covid19DA-Work-DiseaseScreening** class and has this appearance:


1. Gathering Information
Collect Information
1. Require Consent
2. Consent Is Given
3. Define Symptoms
4. Collect Additional ...
5. Change to cancel
CONFIGURE PROCESS
PARALLEL
Gathering Information We...
1. WebChatbot-Gath...
2. Change To Next St...
+ STEP

2. Disease Screening
Disease Prediction
1. Disease Adaptive ...
+ STEP
Provide Information
1. Select Option
2. Collect Contact Info
+ STEP
PARALLEL
Disease Screening WebCh...
1. DiseaseAdaptiveM...
2. WebChatbot-Dise...
3. Change To Next St...
+ STEP

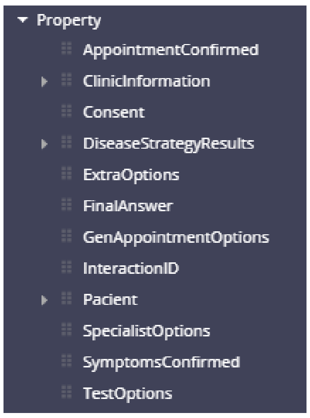
3. Retrieving Clinic Infor...
Retrieve Health Plan
1. Retrieve Health Pl...
+ STEP
Retrieve Calendar
1. Retrieve Available ...
+ STEP

4. Confirming Appointme...
Confirm Appointment
1. Confirm Appoint...
2. Send Email
+ STEP
Update Medical History
1. Update Medical Hi...
+ STEP
PARALLEL
Confirming Appointment ...
1. WebChatbot-Confi...
2. Send Email
3. Update Medical Hi...
4. Wait
+ STEP

5. Confirming Final Diagn...
Diagnosis Confirmation
1. Gather Medical Di...
2. Update Disease A...
+ STEP

Cancelling
+ PROCESS
Cancelling - No Medica...
+ PROCESS
Cancelling - Connectio...
+ PROCESS
+ ALTERNATE STAGE

The DiseaseScreening class has the following list of properties, most of them used as flags or auxiliary properties for the case, and the other described in the previous data model section:

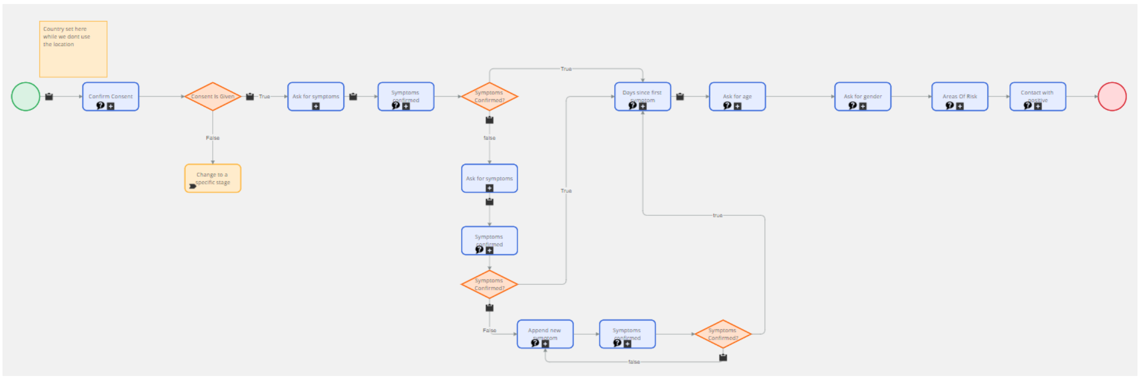

Property
AppointmentConfirmed
ClinicInformation
Consent
DiseaseStrategyResults
ExtraOptions
FinalAnswer
GenAppointmentOptions
InteractionID
Pacient
SpecialistOptions
SymptomsConfirmed
TestOptions

Although it has the "classic" Pega case version, we will focus on the parallel processes of stages one, two and four that we use for the chat bot process. To differ between the executions of processes, we have created the when rule "ChannelIsWebChat" that will execute chat bot processes when true.

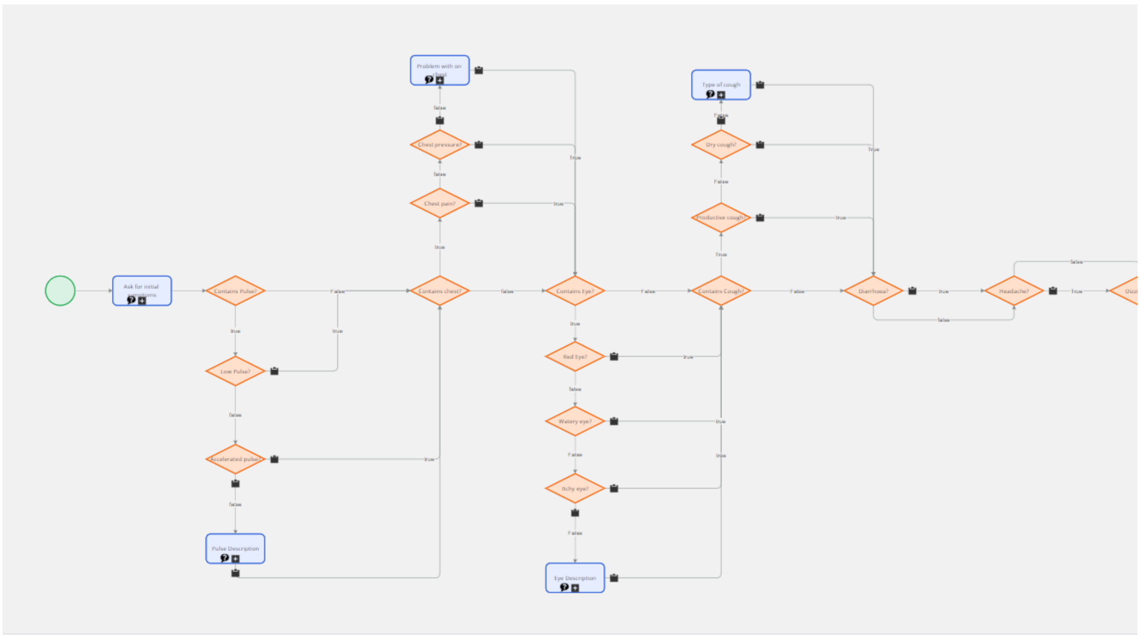
## 3.3. Stage 1. Gathering Information

In this stage, we ask for the consent to proceed with the case, as well as the symptoms and other properties we use for the adaptive model to calculate a first diagnosis.

The main process for the chat bot in this process is the following (and complex) "GatheringInformation_WebChat bot_Conv_1" flow:

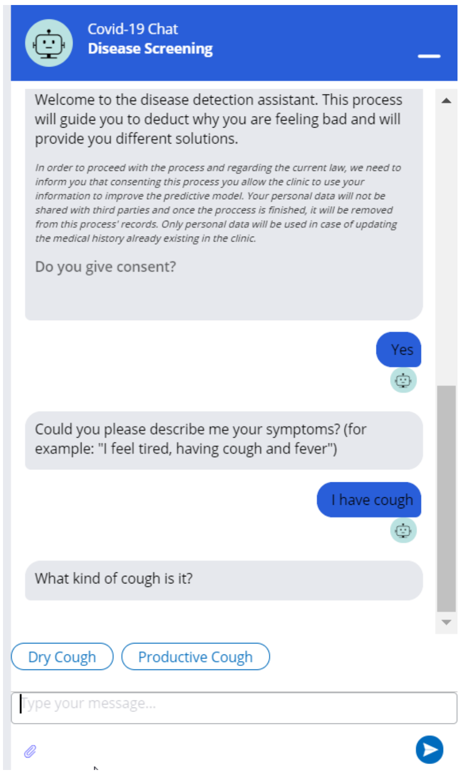


All of the flows related with the chat bot part have the same structure of questions and decision shapes. In "AskForSymptoms" flow, we implement the logic to map the symptoms defined:



The reason to build this on this way was to be able to add final questions shapes in which we can be more accurate on the symptom (see also section 7.1 for a better explanation on this design decision).

In addition, to make the model more accurate we need to differ between kind of symptoms (it is not the same low fever than high fever or dry cough to productive cough). For that reason, for some of the symptoms we have in our model, we try to be as accurate as possible.



Covid-19 Chat
**Disease Screening**

Welcome to the disease detection assistant. This process will guide you to deduct why you are feeling bad and will provide you different solutions.

*In order to proceed with the process and regarding the current law, we need to inform you that consenting this process you allow the clinic to use your information to improve the predictive model. Your personal data will not be shared with third parties and once the proccess is finished, it will be removed from this process' records. Only personal data will be used in case of updating the medical history already existing in the clinic.*

Do you give consent?

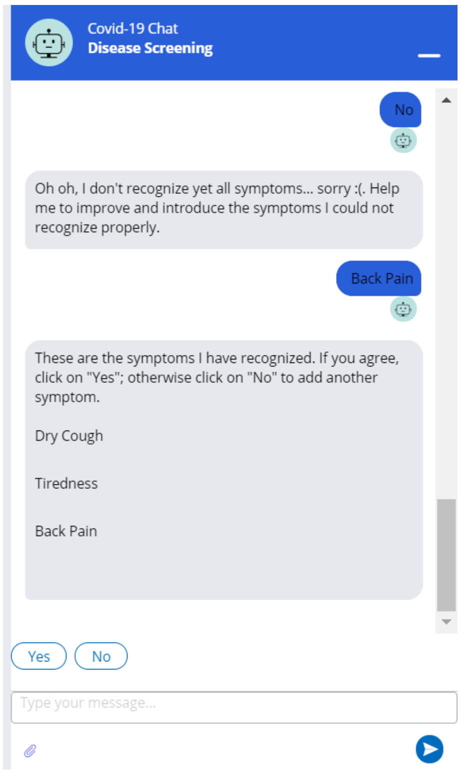
Yes

Could you please describe me your symptoms? (for example: "I feel tired, having cough and fever")

I have cough

What kind of cough is it?

While checking all the symptoms in the flow, we are creating a page list of symptoms (.ListOfSymptoms) in which we append all of them to be able to show them easily on the screen when the patient has just introduced them.



Covid-19 Chat
**Disease Screening**

No

Oh oh, I don't recognize yet all symptoms... sorry :(. Help me to improve and introduce the symptoms I could not recognize properly.

Back Pain

These are the symptoms I have recognized. If you agree, click on "Yes"; otherwise click on "No" to add another symptom.

Dry Cough

Tiredness

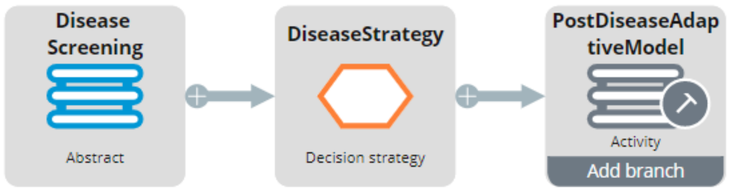
Back Pain

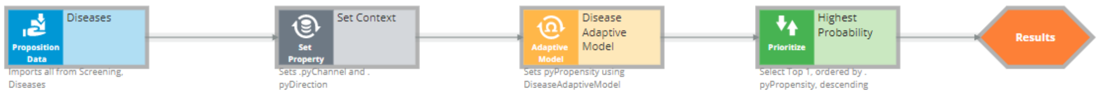
Once the symptoms are confirmed, we proceed with the same questions as in the main flow. This additional data is used by the adaptive model, as well as the list of symptoms, to retrieve the diagnosis with the higher probability:

- **Date of the First Symptom** (validation of date before current date). Actually, the adaptive model uses the number of days since the first symptom, calculated with the date entered and the current date.
- **Age** (validated to be integer between 0 and 120 years).
- **Gender** (only male or female).
- **Areas of Risk**. In this question, we ask if the patient has recently been in one of the countries indicated on the screen. To build this list we use the data page "D_AreasOfRiskList".
- **Contact with Covid-19 Positive**. If the patient has been recently in contact with some person diagnosed as Covid-19 positive.

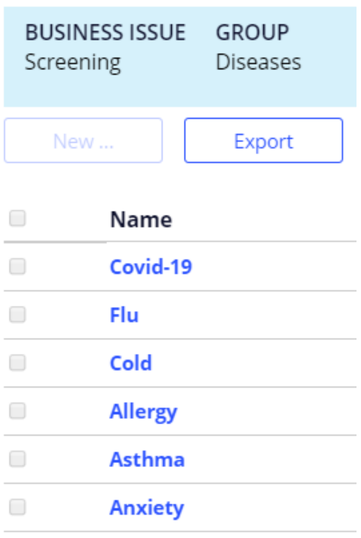
## 3.4. Stage 2. Disease Screening

In this stage, we first run the adaptive model to retrieve which is the disease that more approximates to the symptoms given. The model is used in a Strategy through a Data Flow:




Proposition Data
Diseases
Imports all from Screening, Diseases
Set Property
Set Context
Sets .pyChannel and .pyDirection
Adaptive Model
Disease Adaptive Model
Sets pyPropensity using DiseaseAdaptiveModel
Prioritize
Highest Probability
Select Top 1, ordered by .pyPropensity, descending
Results

The strategy retrieves all the possible diseases as propositions for the "Screening" business issue and "Diseases" group:

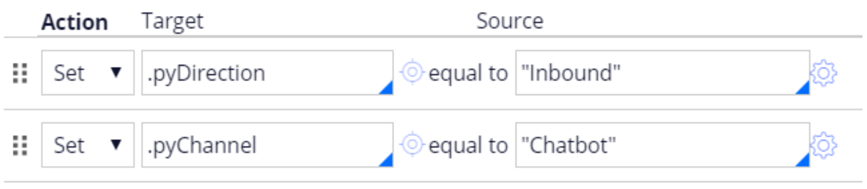


| BUSINESS ISSUE | GROUP |
|---|---|
| Screening | Diseases |

New ... Export

| Name |
|---|
| Covid-19 |
| Flu |
| Cold |
| Allergy |
| Asthma |
| Anxiety |

Then, it sets the context values for the adaptive model:



| Action | Target | | Source |
|---|---|---|---|
| Set | .pyDirection | equal to | "Inbound" |
| Set | .pyChannel | equal to | "Chatbot" |

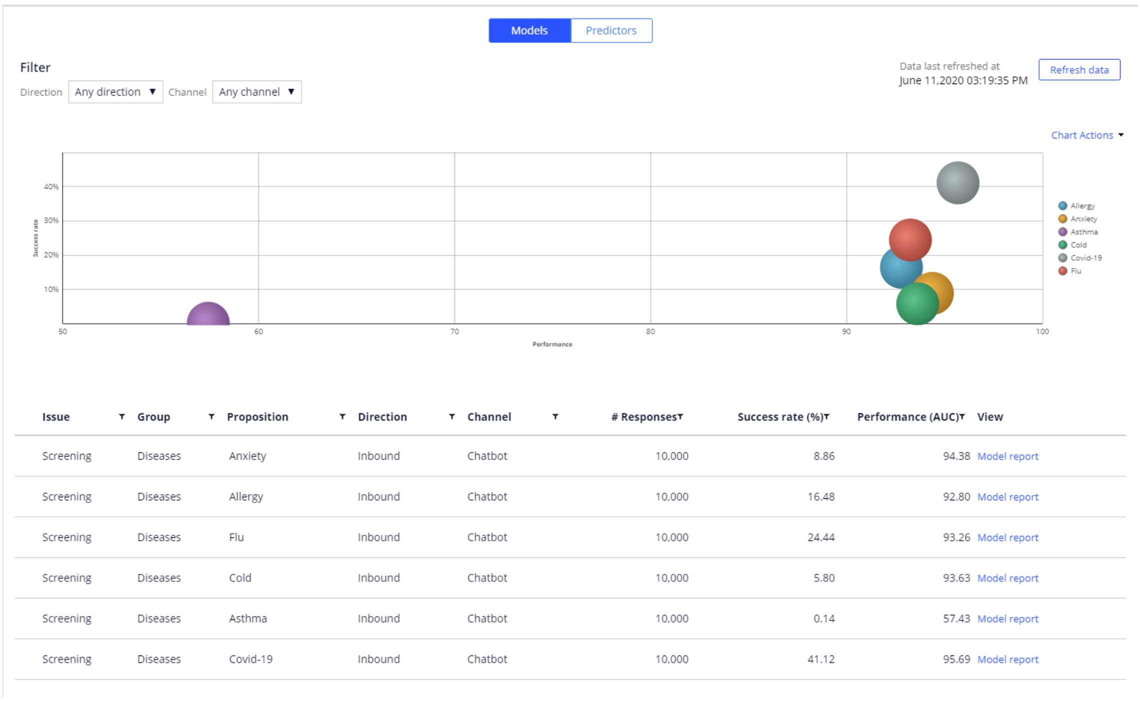
And it calls the “DiseaseAdaptiveModel” rule, in which the following properties are set as predictors of the model:

- .Pacient.Symptoms.[] (all the True/False properties embedded in that page)
- .Pacient.AreaOfRisk
- .Pacient.ContactWithPositive
- .Pacient.DaysSinceFirstSymptoms
- .Pacient.PersonalInformation.Age
- .Pacient.PersonalInformation.Gender

As we only want to give the patient one option as first diagnosis, we have set the strategy to return only the proposition with the higher propensity.

In addition, we will not be able to know if the first diagnosis is correct or not until a doctor provide the final diagnosis in stage 4 (see below). So, in the Data Flow we postpone the response handling by setting the decision strategy configuration to “**Make decision and store data for later response capture**” mode. Using the property .InteractionID, we will be able to map the final output (with outcome as Success or Fail).

Finally, due that the information given by the model is a medical diagnosis, we cannot just create the adaptive model and use it without a training for it to learn, because the initial accurate would be very poor. That is why we have developed a support Java code for generating 10,000 records of random predictor values in a while (with some hardcoded logic to assign the proper disease).



Filter
Direction: Any direction
Channel: Any channel

Data last refreshed at June 11,2020 03:19:35 PM

| Issue | Group | Proposition | Direction | Channel | # Responses | Success rate (%) | Performance (AUC) | View |
|---|---|---|---|---|---|---|---|---|
| Screening | Diseases | Anxiety | Inbound | Chatbot | 10,000 | 8.86 | 94.38 | Model report |
| Screening | Diseases | Allergy | Inbound | Chatbot | 10,000 | 16.48 | 92.80 | Model report |
| Screening | Diseases | Flu | Inbound | Chatbot | 10,000 | 24.44 | 93.26 | Model report |
| Screening | Diseases | Cold | Inbound | Chatbot | 10,000 | 5.80 | 93.63 | Model report |
| Screening | Diseases | Asthma | Inbound | Chatbot | 10,000 | 0.14 | 57.43 | Model report |
| Screening | Diseases | Covid-19 | Inbound | Chatbot | 10,000 | 41.12 | 95.69 | Model report |

## 3.5. Stage 3. Retrieve Clinic Information

On the first process of this stage, we retrieve the hospital information and the patient's health plan information. As this implementation depends directly on each medical institution, we have built a placeholder to simulate this behavior. The idea is:

- First, to obtain basic clinic data as the name or the address of the institution.
- Second, retrieve the health plan information (coverage and guarantees), if the patient has introduced a valid policy number on previous question.

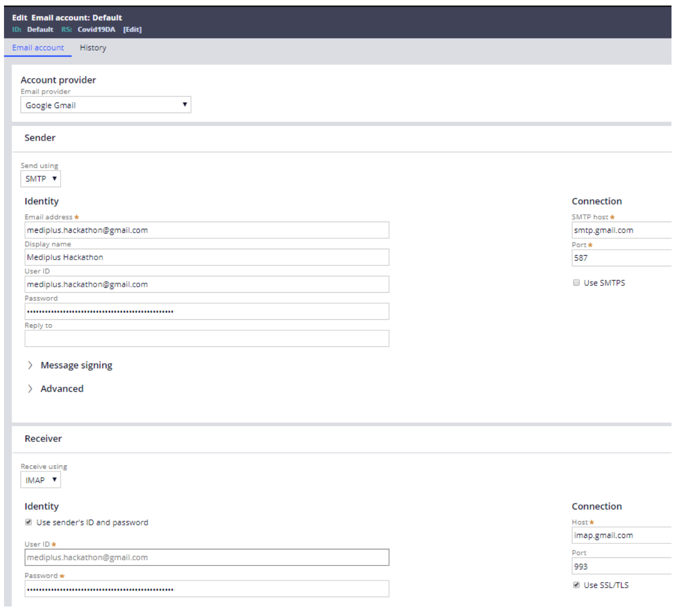
The implementation could be done using a connector in several ways depending on the hospital systems. The ideal would be the use of some **APIs** provided by the medical institution IT services, passing some parameters as the policy number. However, if no APIs are available, we could use **RPA (Robotic Process Automation)**. In case of fail on the connection, a retry mechanism would be executed. This would never block the patient to proceed with the chat bot.

On the second process of the stage, we retrieve the available dates for the option selected previously by the patient. On the same way, a connector could be used with the type of appointment we need to retrieve as a parameter. If there were an error on this connection, again a retry mechanism would be executed. If it were not a transient error, we would recommend the patient to try the chat bot functionality later.

## 3.6. Stage 4. Confirming Appointment

If no error happens, we show to the patient the dates retrieved by the previous connector. To do so, we have created a page list in .ClinicInformation: ".ListOfAvalaibleDates" with all the data retrieved. When the patient confirms the appointment, a summary is shown using all the properties saved during the case and those retrieved from the clinic (as the institution address).

Then, the email shape in the flow sends a summary of the process to the patient email address introduced previously. To avoid issues, we created a Gmail account and configured with it the Default email account in Pega:



Edit Email account: Default
ID: Default RS: Covid19DA [Edit]

Account provider
Email provider: Google Gmail

Sender
Send using: SMTP

Identity
Email address *: mediplus.hackathon@gmail.com
Display name: Mediplus Hackathon
User ID: mediplus.hackathon@gmail.com
Password: ••••••••••••••••••••••••••••••••••••••••••••••
Reply to:

Connection
SMTP host *: smtp.gmail.com
Port *: 587
☐ Use SMTPS

Receiver
Receive using: IMAP

Identity
☑ Use sender's ID and password
User ID *: mediplus.hackathon@gmail.com
Password *: ••••••••••••••••••••••••••••••••••••••••••••••

Connection
Host *: imap.gmail.com
Port: 993
☑ Use SSL/TLS

The idea is to use the email account of the medical institution and not one of Gmail (see Installation and Customization section below).

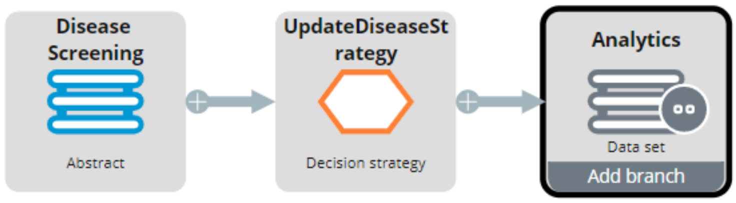
After the email is sent, we have added a placeholder just in case we could update any information on the hospital systems (such as the appointment confirmation, the medical history of the patient, etc.). This could be done using a specific connector as in the previous placeholders (through APIs or RPA, depending on the hospital systems).

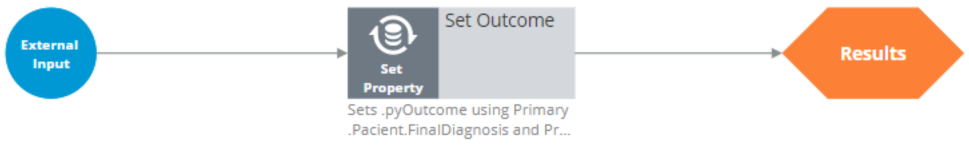
Finally, a wait shape is added and set to 6 minutes. The reason is to avoid the case to proceed or to show any other message to the patient we do not want to show. As the chat bot is set to have a timeout of 5 minutes, we make sure that the case is proceed to next stage only when the patient cannot have access.

## 3.7. Stage 5. Confirming Final Diagnosis

We have created this stage to retrieve from the hospital the final diagnosis entered by the doctor who has treated the patient on the appointment. The final diagnosis will be one of the diseases we use on our model or "Other" if no one matches.

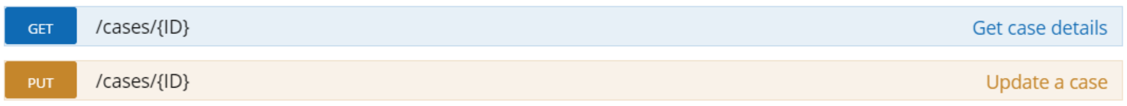
After updating the final diagnosis, we set the response in the adaptive model to improve its accuracy and reliability. Similar as in stage 2, we run a Data Flow calling a Strategy:




External Input
Set Property
Set Outcome
Sets .pyOutcome using Primary .Pacient.FinalDiagnosis and Pr...
Results

We recover the previous proposition offered as first diagnosis by setting in the Data Flow the decision strategy configuration to "**Capture response from previous decision by Interaction ID**" mode. Finally, we set the outcome (Success or Fail) to tell the model if it was right and let it learn.

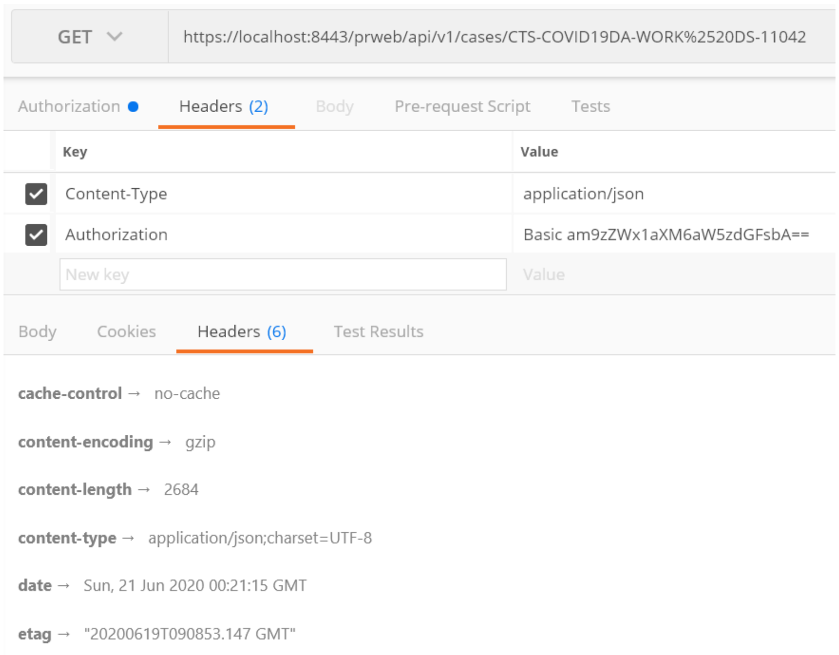
The selected way to integrate this stage with the medical institution systems is through the **Pega API**, invoking the "SetFinalDiagnosis" local action placed in this final stage, that executes the same logic as the normal flow of the stage. The integration should use these two calls:



| | | |
|---|---|---|
| GET | /cases/{ID} | Get case details |
| PUT | /cases/{ID} | Update a case |

Both calls need a Basic Authentication to connect with the Pega server, that must be running in HTTPS mode (see Installation and Customization below).

The first call (GET) is only used to retrieve the **ETag** header:

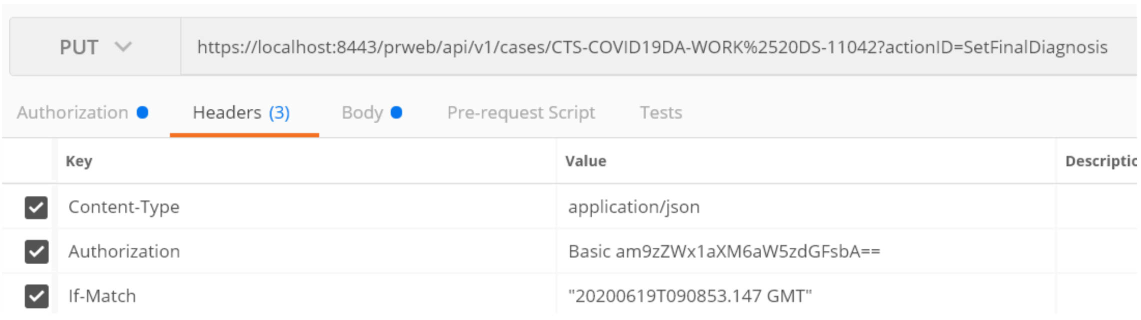


```
GET https://localhost:8443/prweb/api/v1/cases/CTS-COVID19DA-WORK%2520DS-11042
```

| Key | Value |
|---|---|
| Content-Type | application/json |
| Authorization | Basic am9zZWx1aXM6aW5zdGFsbA== |

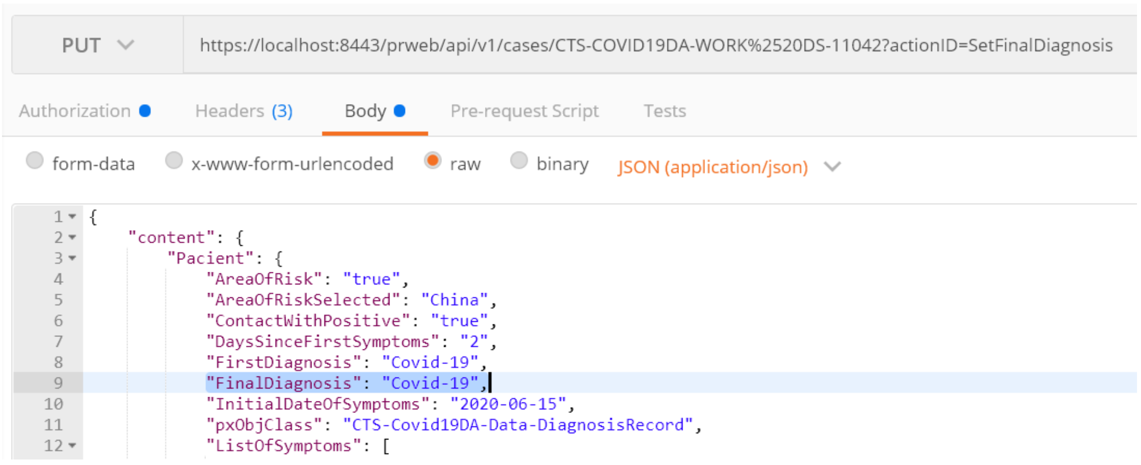
```
cache-control → no-cache
content-encoding → gzip
content-length → 2684
content-type → application/json;charset=UTF-8
date → Sun, 21 Jun 2020 00:21:15 GMT
etag → "20200619T090853.147 GMT"
```

The second call (PUT) uses the ETag in the **If-Match** header and updates the case with the final diagnosis, invoking the "SetFinalDiagnosis" local action placed in this final stage:



```
PUT https://localhost:8443/prweb/api/v1/cases/CTS-COVID19DA-WORK%2520DS-11042?actionID=SetFinalDiagnosis
```

| Key | Value | Descriptic |
|---|---|---|
| Content-Type | application/json | |
| Authorization | Basic am9zZWx1aXM6aW5zdGFsbA== | |
| If-Match | "20200619T090853.147 GMT" | |



```
PUT https://localhost:8443/prweb/api/v1/cases/CTS-COVID19DA-WORK%2520DS-11042?actionID=SetFinalDiagnosis
```

```
{
    "content": {
        "Pacient": {
            "AreaOfRisk": "true",
            "AreaOfRiskSelected": "China",
            "ContactWithPositive": "true",
            "DaysSinceFirstSymptoms": "2",
            "FirstDiagnosis": "Covid-19",
            "FinalDiagnosis": "Covid-19",
            "InitialDateOfSymptoms": "2020-06-15",
            "pxObjClass": "CTS-Covid19DA-Data-DiagnosisRecord",
            "ListOfSymptoms": [
```

## 4. Custom Reports

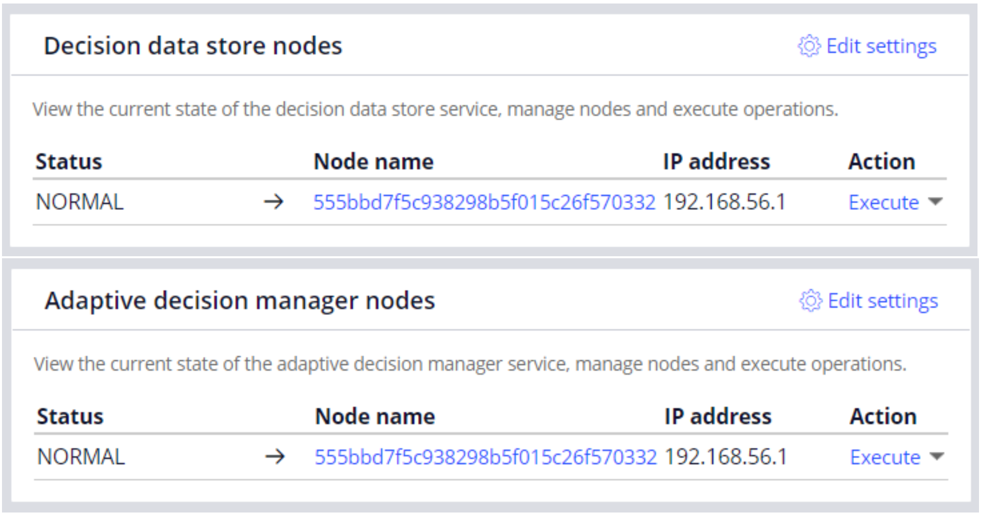
In addition to the OOTB reports provided by Pega, we have configured some custom reports in case the hospital wanted to use them. For example, one of them is the "Diagnosis per day in last 30 days". This report may be useful to detect the spread of a disease (for example we can see an increase or a decrease of Covid-19 diagnosis). Indeed, the report can be adjusted to a different period (15 days instead of 30 days, etc.).

## 5. Installation and Customization

In order to use this application in a real situation for a real medical institution, we must make some adjustments during the installation and configuration.

- We need at least one node running in **DDS (Decision Data Store)** and **ADM (Adaptive Decision Manager)** modes. To configure the nodes in such a way, we need to run the JVM adding those modes to the NodeType parameter. For example:

  **-DNodeType=**Stream,BackgroundProcessing,WebUser,Search,**DDS,ADM**



Decision data store nodes

View the current state of the decision data store service, manage nodes and execute operations.

| Status | Node name | IP address | Action |
|---|---|---|---|
| NORMAL | → 555bbd7f5c938298b5f015c26f570332 | 192.168.56.1 | Execute |

Adaptive decision manager nodes

View the current state of the adaptive decision manager service, manage nodes and execute operations.

| Status | Node name | IP address | Action |
|---|---|---|---|
| NORMAL | → 555bbd7f5c938298b5f015c26f570332 | 192.168.56.1 | Execute |

- We need to configure the server to run in **HTTPS** mode, so we can call the **Pega API** when updating the case in the last stage.
- Update the Default email account with the one you want to use for sending the confirmation messages.

Obviously, we must develop the necessary integrations with the clinic to replace the placeholders in the case life cycle. As mentioned earlier, the way of doing this depends on the hospital IT systems and it could be done through APIs or a RPA solution. The complete list of placeholders is as follows:

- Retrieve Clinic Information
- Retrieve Health Plan
- Retrieve Available Dates
- Update Medical History

Finally, we must also integrate the Pega API calls to update the final diagnosis.

## 6. What was Left Behind

As in many projects, we had to put aside some ideas that we wanted to implement. The inclusion of these ideas could be done with a larger team or more time to develop a final version of this application:

### 6.1. List of Risk Areas

We could use a RPA solution to retrieve and store in our application the list of countries considered as areas of risk for each country. We could do this by accessing the official health websites of the different countries, maybe once per week.

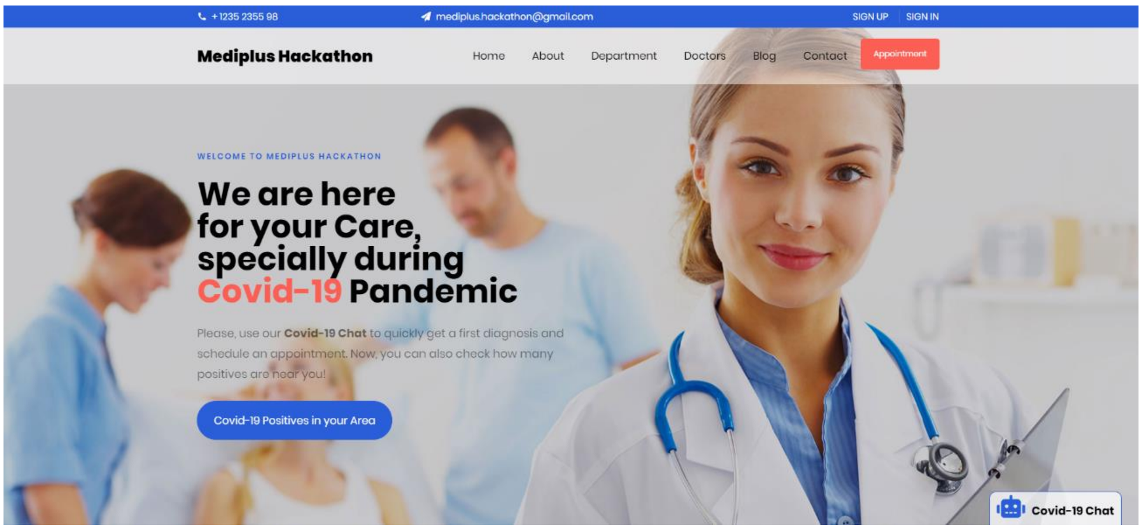
### 6.2. Geolocation

Once the consent was given by the patient, we would have retrieved the latitude and longitude of his/her position at that time. Using these two properties and configuring a connector to the Google API, we could have obtained the country from which this patient is interacting with the chat bot.

Knowing the origin country, we would have retrieved and populate in the case the list of countries considered as areas of risk.

### 6.3. Covid-19 Positives in your Area

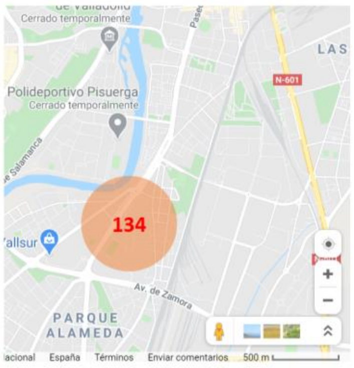
In addition to the chat bot, we could embed a button on the website of the medical institution to enable the "Covid-19 Positives in your Area" functionality.



We are here for your Care, specially during Covid-19 Pandemic

Please, use our **Covid-19 Chat** to quickly get a first diagnosis and schedule an appointment. Now, you can also check how many positives are near you!

We could show a map of Covid-19 positives that have been diagnosed by the hospital in a predefined radius of 1km, 2km etc. from the current position.



For a better performance, the required data should be anonymously stored in our application. The records could be populated daily from the hospital database by an API or RPA connector.

# 7. What to Improve

## 7.1. Chat bot

The initial idea was to build a chat bot based on NLP through the Pega IVA (Intelligent Virtual Assistant), but we decided to build a different solution for the chat bot, because of lack of time and knowledge. We had two main topics to learn and put into operation: the chat bot and the decisioning adaptive model.

We decided to prevail the adaptive model part. So, as the predictors need to be filled for the adaptive model to work, we proceed with a static solution for the chat bot.

In this first approach of the solution, the chat bot interaction is based on flows using questions and decision shapes, with different behavior depending on the answers and the symptoms. In addition to this, the code used in the flows is huge and hard to maintain and update. Using NLP, we presume this would not be the case, because the model would learn what to ask depending on the symptoms.

## 7.2. Adaptive Model Training

Currently, the adaptive model has been trained with 10,000 records of random predictor values (with some hardcoded logic to assign the proper disease) generated by a simple Java code. Obviously, the real model must be trained with real data, probably obtained from the database of the medical institution.

## 7.3. Error Handling

As we mentioned earlier, we know that we need to improve error handling throughout the entire application. Mainly when implementing the real connectors with the medical institution instead of the placeholders, but also in sending emails or retrieving the final diagnosis.

## 7.4. Custom Reports

In addition to the OOTB reports provided by Pega, we have configured some custom reports in case the hospital wanted to use them. However, we are sure that we should create more to get interesting statistics that could help to improve the customer satisfaction or increase the ROI of the medical institution.

### 7.5. More Symptoms and Diseases

We have created this solution based on a limited set of symptoms and diseases (see appendix A and B), but it could be easily improved by adding more symptoms (for example high or low blood pressure, back pain, etc.) and diseases (such as dengue).

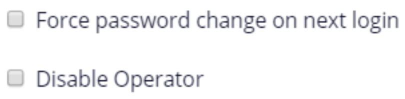
### 7.6. Protection Data Law

We need to accomplish with the Data Protection Law of each country, for example the **GDPR (General Data Protection Regulation)** in the EU (European Union). So, the initial message could differ between the countries using the application. What we have written is a generic message that should be changed when developing the final application.

## 8. Testing the Application

Please, follow the next steps to install and test the application:

- Update the Pega environment as described in the Installation and Customization section above:
  - Configure nodes for DDS and ADM (or Universal)
  - Enable HTTPS
  - Update the Default email account. For testing purposes you can just use the one already configured: mediplus.hackathon@gmail.com
- Import the Disease Screening application zip file:
  - Covid19DA_010101_20200622T121354_GMT.zip
- Add the following Access Groups to your Operator:
  - Covid19AD:Developers
  - Covid19AD:Authors
  - Covid19AD:Users
- Switch to “Covid-19 Detection Assistant (Cosmos Developers)” application.
- Open the “DiseaseStrategy” Strategy rule and run it.
- Open the “Decisioning > Model Management” Landing Page, and train the six adaptive models created (one per proposition/disease) with the 10,000 records in the training files provided with this document:
  - Allergy.csv
  - Anxiety.csv
  - Asthma.csv
  - Cold.csv
  - Covid19.csv
  - Flu.csv
- Open the “ChatBotOperator” Operator ID rule and be sure that the Security tab is configuration this way:



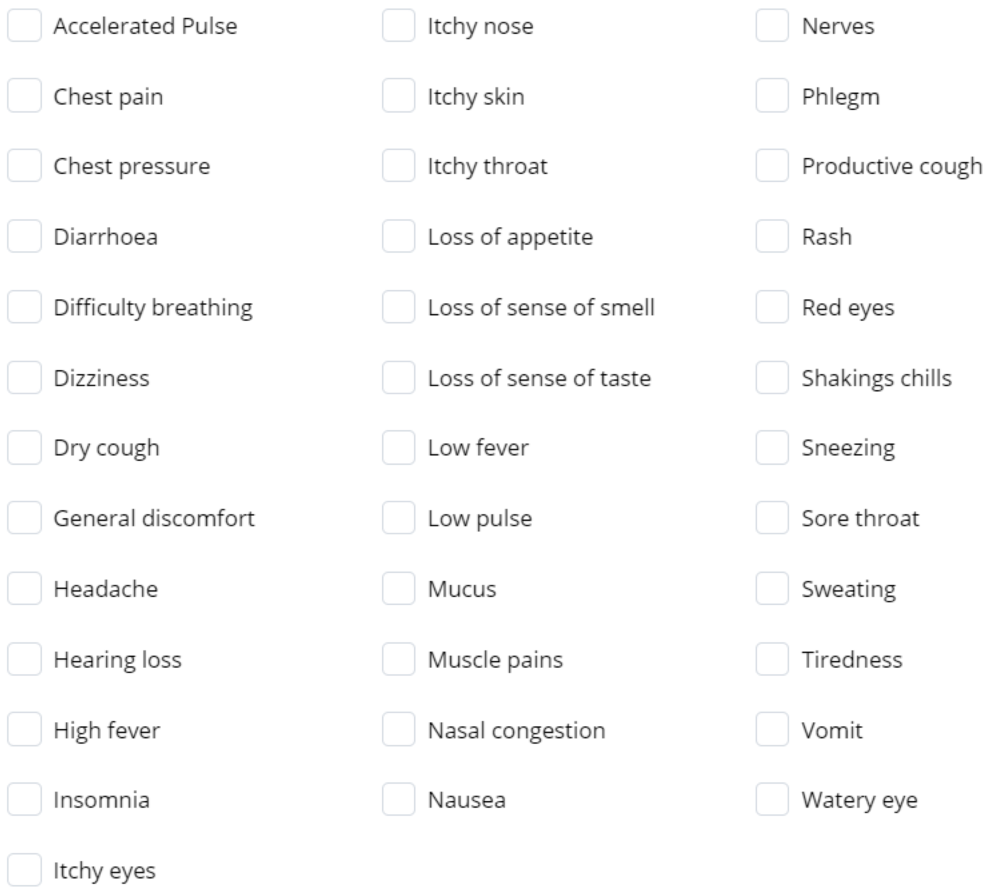
☐ Force password change on next login

☐ Disable Operator

- Open the “Channels and interfaces” Landing Page, select the “Disease Screening” Chatbot and try it in the preview console (for example by following the Case of Use described in section 2 of this document).
    - Optional: you can change the mashup URL with the one used by your Pega environment, and embed the code in a mockup medical institution website (like the “Mediplus Hackathon” website that we have used in the examples and video recording).
- Once the case is finished in the Chatbot, you can open it to simulate the final diagnosis entered by the doctor. The case will be in the last stage (Confirming Final Diagnosis), just select one of the diseases to complete it and the adaptive model will be updated with the response automatically.
    - Optional: you can also do this final step through a couple of Pega API calls, as described in section 3.7 of this document. First call (GET) is only used to retrieve the ETag header and the second call (PUT) uses the ETag in the If-Match header and updates the case with the final diagnosis, invoking the “SetFinalDiagnosis” local action placed in this final stage.

## Appendix A: Symptoms

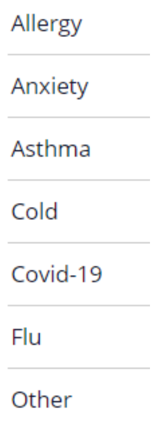
List of symptoms detected by the chat bot and used by the adaptive model to retrieve the first diagnosis:



| | | |
|---|---|---|
| ☐ Accelerated Pulse | ☐ Itchy nose | ☐ Nerves |
| ☐ Chest pain | ☐ Itchy skin | ☐ Phlegm |
| ☐ Chest pressure | ☐ Itchy throat | ☐ Productive cough |
| ☐ Diarrhoea | ☐ Loss of appetite | ☐ Rash |
| ☐ Difficulty breathing | ☐ Loss of sense of smell | ☐ Red eyes |
| ☐ Dizziness | ☐ Loss of sense of taste | ☐ Shakings chills |
| ☐ Dry cough | ☐ Low fever | ☐ Sneezing |
| ☐ General discomfort | ☐ Low pulse | ☐ Sore throat |
| ☐ Headache | ☐ Mucus | ☐ Sweating |
| ☐ Hearing loss | ☐ Muscle pains | ☐ Tiredness |
| ☐ High fever | ☐ Nasal congestion | ☐ Vomit |
| ☐ Insomnia | ☐ Nausea | ☐ Watery eye |
| ☐ Itchy eyes | | |

## Appendix B: Diseases

List of diseases detected by the adaptive model (plus “Other” in case that the final diagnosis is not any of them):



Allergy

Anxiety

Asthma

Cold

Covid-19

Flu

Other