

# Documento del sistema

# Aplicaciones Industriales

Andrés Tena De Tena

Javier Albaráñez Martínez

## Índice

|      |                                  |   |
|------|----------------------------------|---|
| I.   | Introducción .....               | 1 |
| II.  | Despliegue.....                  | 6 |
| III. | Funcionamiento del sistema ..... | 8 |

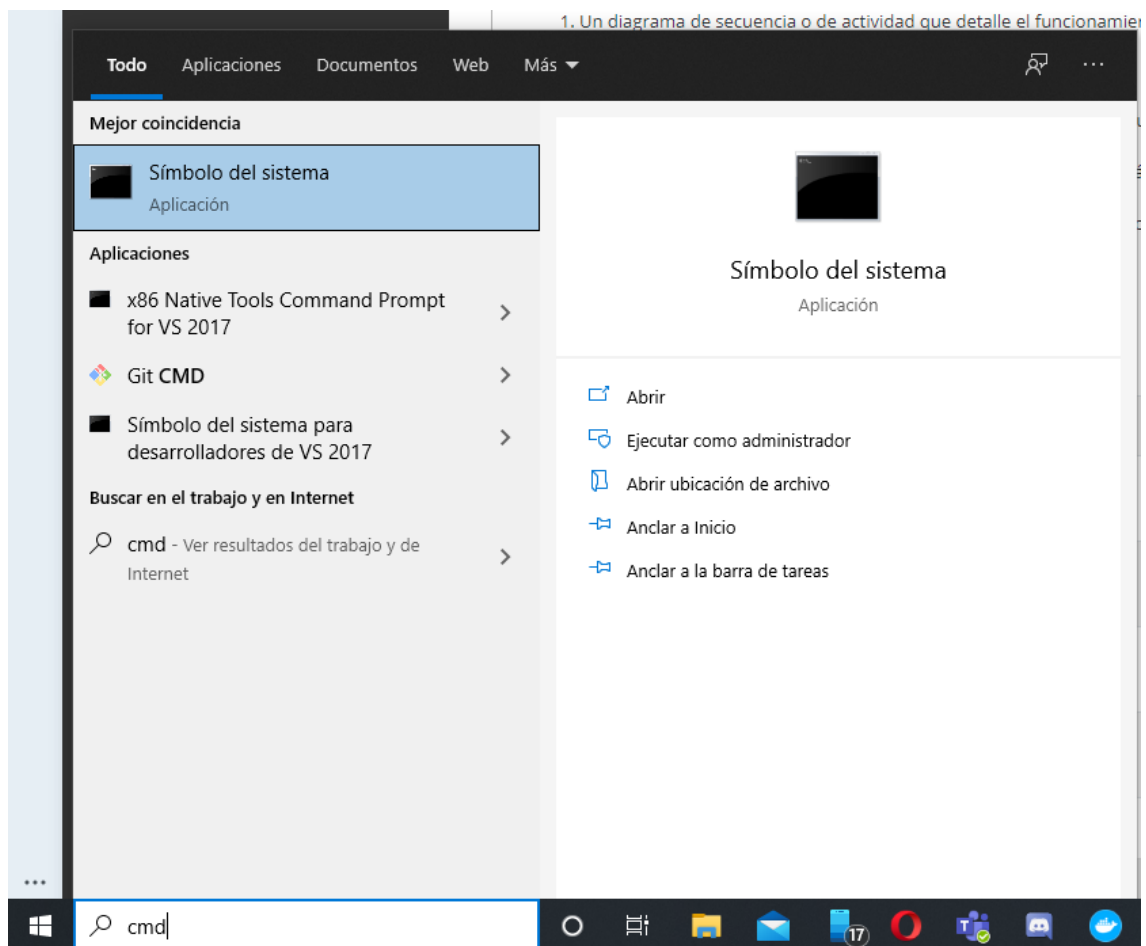
# I. Introducción

Se ha desarrollado una aplicación capaz de identificar y numerar el porcentaje de imperfecciones que posee una pieza de madera. Además, esta aplicación se ha subido a una imagen Docker desde la cual se puede ejecutar sin problema, simplemente ejecutando un archivo .java. A continuación, se detallan los pasos a seguir para instalar la imagen Docker en Windows y ejecutar la aplicación:

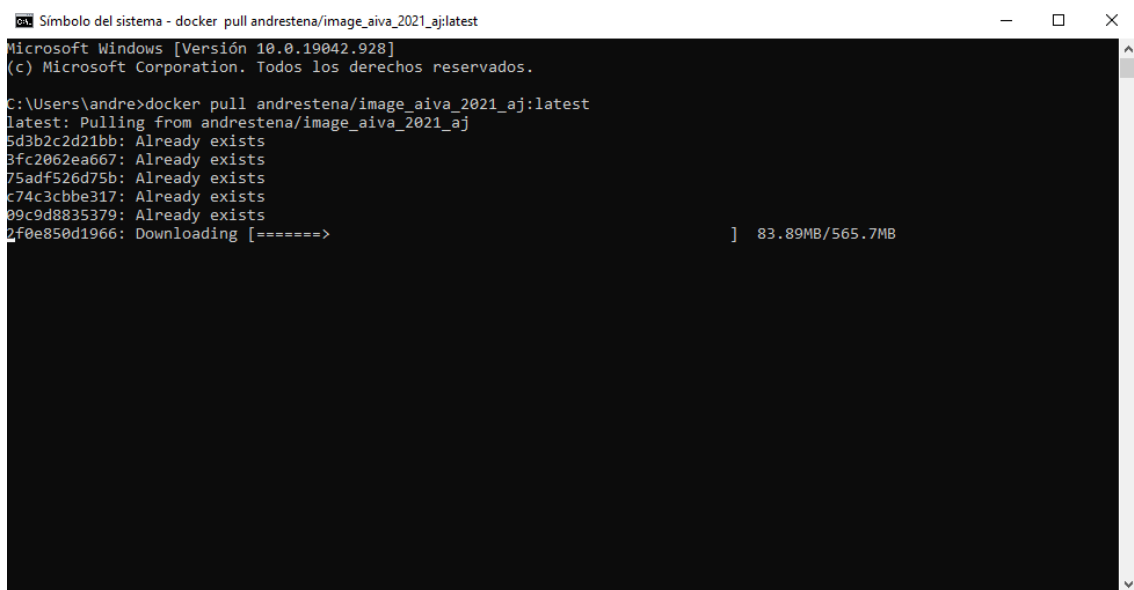
Enlace a GitHub:

[https://github.com/AlbaranezJavier/AIVA\\_2021\\_AJ](https://github.com/AlbaranezJavier/AIVA_2021_AJ)

Paso 1: Abrir la cmd de Windows.



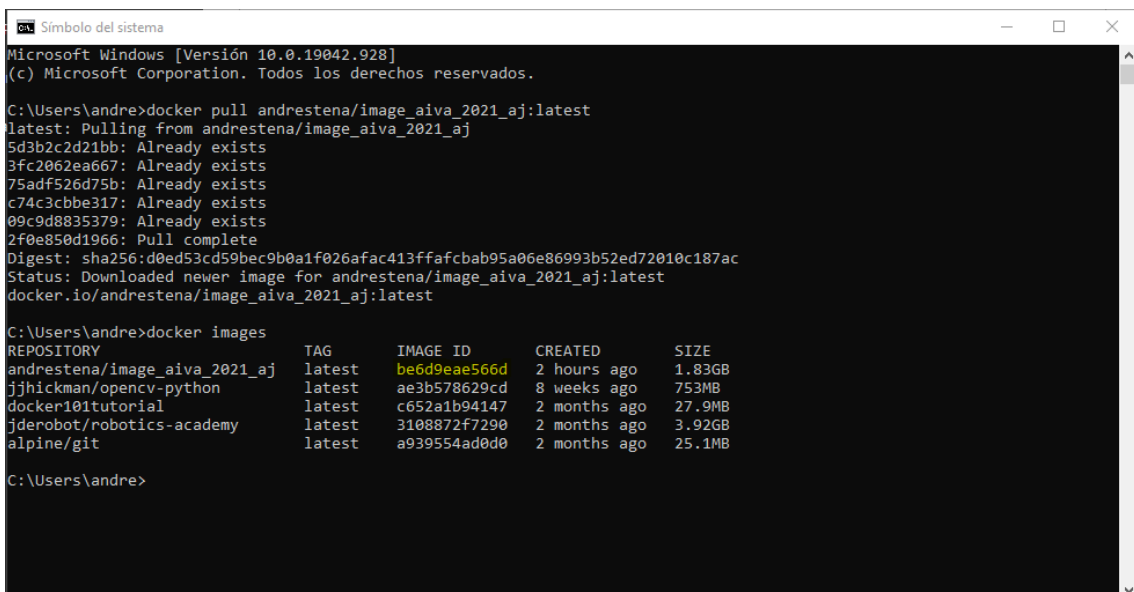
Paso 2: Ejecutar el comando `docker pull andrestena/image_aiva_2021_aj:latest` en la cmd y esperar a que se descargue la imagen.



```
Símbolo del sistema - docker pull andrestena/image_aiva_2021_aj:latest
Microsoft Windows [Versión 10.0.19042.928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\andre>docker pull andrestena/image_aiva_2021_aj:latest
latest: Pulling from andrestena/image_aiva_2021_aj
5d3b2c2d21bb: Already exists
3fc2062ea667: Already exists
75adf526d75b: Already exists
c74c3cbb317: Already exists
09c9d8835379: Already exists
2f0e850d1966: Downloading [=====>] 83.89MB/565.7MB
```

Paso 3: Ejecutar el comando `docker images` en la cmd, una vez ejecutado este comando se deberá fijar en la IMAGE ID del repositorio `andrestena/image_aiva_2021_aj`.



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19042.928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\andre>docker pull andrestena/image_aiva_2021_aj:latest
latest: Pulling from andrestena/image_aiva_2021_aj
5d3b2c2d21bb: Already exists
3fc2062ea667: Already exists
75adf526d75b: Already exists
c74c3cbb317: Already exists
09c9d8835379: Already exists
2f0e850d1966: Pull complete
Digest: sha256:d0ed53cd59bec9b0a1f026afac413ffafcbab95a06e86993b52ed72010c187ac
Status: Downloaded newer image for andrestena/image_aiva_2021_aj:latest
docker.io/andrestena/image_aiva_2021_aj:latest

C:\Users\andre>docker images
REPOSITORY              TAG          IMAGE ID       CREATED        SIZE
andrestena/image_aiva_2021_aj  latest      be6d9eae566d   2 hours ago    1.83GB
jjhickman/opencv-python  latest      ae3b578629cd   8 weeks ago    753MB
docker101tutorial        latest      c652a1b94147   2 months ago   27.9MB
jderobot/robotics-academy latest      3108872f7290   2 months ago   3.92GB
alpine/git               latest      a939554ad0d0   2 months ago   25.1MB

C:\Users\andre>
```

Paso 4: Ejecutar el comando *docker create -it ID bash*, en ID se tendrá que poner el IMAGE ID obtenido en el paso anterior.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19042.928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\andre>docker pull andrestena/image_aiva_2021_aj:latest
latest: Pulling from andrestena/image_aiva_2021_aj
5d3b2c2d21bb: Already exists
3fc2062ea667: Already exists
75adf526d75b: Already exists
c74c3cbe317: Already exists
09c9d8835379: Already exists
2f0e850d1966: Pull complete
Digest: sha256:d0ed53cd59bec9b0a1f026afac413ffa6cbab95a06e86993b52ed72010c187ac
Status: Downloaded newer image for andrestena/image_aiva_2021_aj:latest
docker.io/andrestena/image_aiva_2021_aj:latest

C:\Users\andre>docker images
REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
andrestena/image_aiva_2021_aj  latest      be6d9eae566d   2 hours ago   1.83GB
fjhickman/opencv-python  latest      ae3b578629cd   8 weeks ago   753MB
docker101tutorial      latest      c652a1b94147   2 months ago   27.9MB
jderobot/robotics-academy  latest      3108872f7290   2 months ago   3.92GB
alpine/git              latest      a939554ad0d0   2 months ago   25.1MB

C:\Users\andre>docker create -it be6d9eae566d bash
e4f2cd95f178cf9c4ba0ddbea2b28c89c8dfc3f33d1a7e8880843122cb709ce2

C:\Users\andre>
```

Paso 5: Comprobar que el contenedor ha sido creado, para ello ejecutar el comando *docker container ls -a*.

```
Símbolo del sistema
Digest: sha256:d0ed53cd59bec9b0a1f026afac413ffa6cbab95a06e86993b52ed72010c187ac
Status: Downloaded newer image for andrestena/image_aiva_2021_aj:latest
docker.io/andrestena/image_aiva_2021_aj:latest

C:\Users\andre>docker images
REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
andrestena/image_aiva_2021_aj  latest      be6d9eae566d   2 hours ago   1.83GB
fjhickman/opencv-python  latest      ae3b578629cd   8 weeks ago   753MB
docker101tutorial      latest      c652a1b94147   2 months ago   27.9MB
jderobot/robotics-academy  latest      3108872f7290   2 months ago   3.92GB
alpine/git              latest      a939554ad0d0   2 months ago   25.1MB

C:\Users\andre>docker create -it be6d9eae566d bash
e4f2cd95f178cf9c4ba0ddbea2b28c89c8dfc3f33d1a7e8880843122cb709ce2

C:\Users\andre>docker container ls -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
e4f2cd95f178  be6d9eae566d  "bash"                  About a minute ago   Created              exciting_lehmann
cb455f21934c  ae3b578629cd  "bash"                  15 hours ago       Exited (0) 12 minutes ago   epic_shockley
1072d838f7fe  jderobot/robotics-academy  "/ros_entrypoint.sh ..." 2 months ago       Exited (137) 6 weeks ago    crazy_albattani
ebda0a387ead  docker101tutorial  "/docker-entrypoint. ..." 2 months ago       Exited (0) 2 months ago    docker-tutorial
868b90b439ee  alpine/git      "git clone https://g..." 2 months ago       Exited (128) 2 hours ago    repo

C:\Users\andre>
```

Paso 6: A continuación, se utilizará el CONTAINER ID señalado para iniciar nuestro contenedor. Para ello ejecutar *docker start CONTAINER ID*.

```

ca Símbolo del sistema
Digest: sha256:d0ed53cd59bec9b0a1f026afac413ffafcbab95a06e86993b52ed72010c187ac
Status: Downloaded newer image for andrestena/image_aiva_2021_aj:latest
docker.io/andrestena/image_aiva_2021_aj:latest

C:\Users\andre>docker images
REPOSITORY          TAG         IMAGE ID      CREATED       SIZE
andrestena/image_aiva_2021_aj  latest     be6d9eae566d  2 hours ago  1.83GB
jjhickman/opencv-python  latest     ae3b578629cd  8 weeks ago  753MB
docker101tutorial      latest     c652a1b94147  2 months ago  27.9MB
jderobot/robotics-academy  latest     3108872f7290  2 months ago  3.92GB
alpine/git             latest     a939554ad0d0  2 months ago  25.1MB

C:\Users\andre>docker create -it be6d9eae566d bash
e4f2cd95f178cf9c4ba0ddbea2b28c89c8dfc3f33d1a7e8880843122cb709ce2

C:\Users\andre>docker container ls -a
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS              PORTS
e4f2cd95f178   be6d9eae566d   "bash"                  About a minute ago   Created
cb455f21934c   ae3b578629cd   "bash"                  15 hours ago       Exited (0) 12 minutes ago
1972d838f7fe   jderobot/robotics-academy  "/ros_entrypoint.sh ..."  2 months ago       Exited (137) 6 weeks ago
ebda0a387ead   docker101tutorial  "/docker-entrypoint..."  2 months ago       Exited (0) 2 months ago
868b90b439ee   alpine/git      "git clone https://g..."  2 months ago       Exited (128) 2 hours ago

C:\Users\andre>docker start e4f2cd95f178
e4f2cd95f178

C:\Users\andre>

```

Paso 7: A continuación, se deberá vincular al contenedor ejecutado. Para ello ejecutar *docker attach CONTAINER ID*.

```

ca root@e4f2cd95f178: /opencv

andrestena/image_aiva_2021_aj  latest     be6d9eae566d  2 hours ago  1.83GB
jjhickman/opencv-python  latest     ae3b578629cd  8 weeks ago  753MB
docker101tutorial      latest     c652a1b94147  2 months ago  27.9MB
jderobot/robotics-academy  latest     3108872f7290  2 months ago  3.92GB
alpine/git             latest     a939554ad0d0  2 months ago  25.1MB

C:\Users\andre>docker create -it be6d9eae566d bash
e4f2cd95f178cf9c4ba0ddbea2b28c89c8dfc3f33d1a7e8880843122cb709ce2

C:\Users\andre>docker container ls -a
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS              PORTS
e4f2cd95f178   be6d9eae566d   "bash"                  About a minute ago   Created
cb455f21934c   ae3b578629cd   "bash"                  15 hours ago       Exited (0) 12 minutes ago
1972d838f7fe   jderobot/robotics-academy  "/ros_entrypoint.sh ..."  2 months ago       Exited (137) 6 weeks ago
ebda0a387ead   docker101tutorial  "/docker-entrypoint..."  2 months ago       Exited (0) 2 months ago
868b90b439ee   alpine/git      "git clone https://g..."  2 months ago       Exited (128) 2 hours ago

C:\Users\andre>docker start e4f2cd95f178
e4f2cd95f178

C:\Users\andre>docker attach e4f2cd95f178
root@e4f2cd95f178: /opencv#

```

Paso 8: Una vez dentro del contenedor, para ejecutar la aplicación será necesario ejecutar el siguiente comando `java /tmp/AIVA_2021_AJ/Java/src/co/ortizol/Main.java`.

```

root@e4f2cd95f178: /opencv
andreestena/image_aiva_2021_aj latest be6d9eae566d 2 hours ago 1.83GB
jjhickman/opencv-python latest ae3b578629cd 8 weeks ago 753MB
docker101tutorial latest c652a1b94147 2 months ago 27.9MB
jderobot/robotics-academy latest 3108872f7290 2 months ago 3.92GB
alpine/git latest a939554ad0d0 2 months ago 25.1MB

C:\Users\andre>docker create -it be6d9eae566d bash
e4f2cd95f178cf9c4ba0ddbea2b28c89c8dfc3f33d1a7e8880843122cb709ce2

C:\Users\andre>docker container ls -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
e4f2cd95f178 be6d9eae566d "bash" About a minute ago Created
cb455f21934c ae3b578629cd "bash" 15 hours ago Exited (0) 12 minutes ago
1972d838f7fe jderobot/robotics-academy "/ros_entrypoint.sh ..." 2 months ago Exited (137) 6 weeks ago
ebda0a387ead docker101tutorial "/docker-entrypoint..." 2 months ago Exited (0) 2 months ago
868b90b439ee alpine/git "git clone https://g..." 2 months ago Exited (128) 2 hours ago

C:\Users\andre>docker start e4f2cd95f178
e4f2cd95f178

C:\Users\andre>docker attach e4f2cd95f178
root@e4f2cd95f178:/opencv# java /tmp/AIVA_2021_AJ/Java/src/co/ortizol/Main.java
Do you want to analyze a Train or Test image?

```

Paso 9: Finalmente el algoritmo nos preguntará donde se encuentra la imagen que quieres analizar si en Train o en Test, además del ID de esta.

```

root@e4f2cd95f178: /opencv
andreestena/image_aiva_2021_aj latest be6d9eae566d 2 hours ago 1.83GB
jjhickman/opencv-python latest ae3b578629cd 8 weeks ago 753MB
docker101tutorial latest c652a1b94147 2 months ago 27.9MB
jderobot/robotics-academy latest 3108872f7290 2 months ago 3.92GB
alpine/git latest a939554ad0d0 2 months ago 25.1MB

C:\Users\andre>docker create -it be6d9eae566d bash
e4f2cd95f178cf9c4ba0ddbea2b28c89c8dfc3f33d1a7e8880843122cb709ce2

C:\Users\andre>docker container ls -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
e4f2cd95f178 be6d9eae566d "bash" About a minute ago Created exciting_lehmann
cb455f21934c ae3b578629cd "bash" 15 hours ago Exited (0) 12 minutes ago epic_shockley
1972d838f7fe jderobot/robotics-academy "/ros_entrypoint.sh ..." 2 months ago Exited (137) 6 weeks ago crazy_albattani
ebda0a387ead docker101tutorial "/docker-entrypoint..." 2 months ago Exited (0) 2 months ago docker-tutorial
868b90b439ee alpine/git "git clone https://g..." 2 months ago Exited (128) 2 hours ago repo

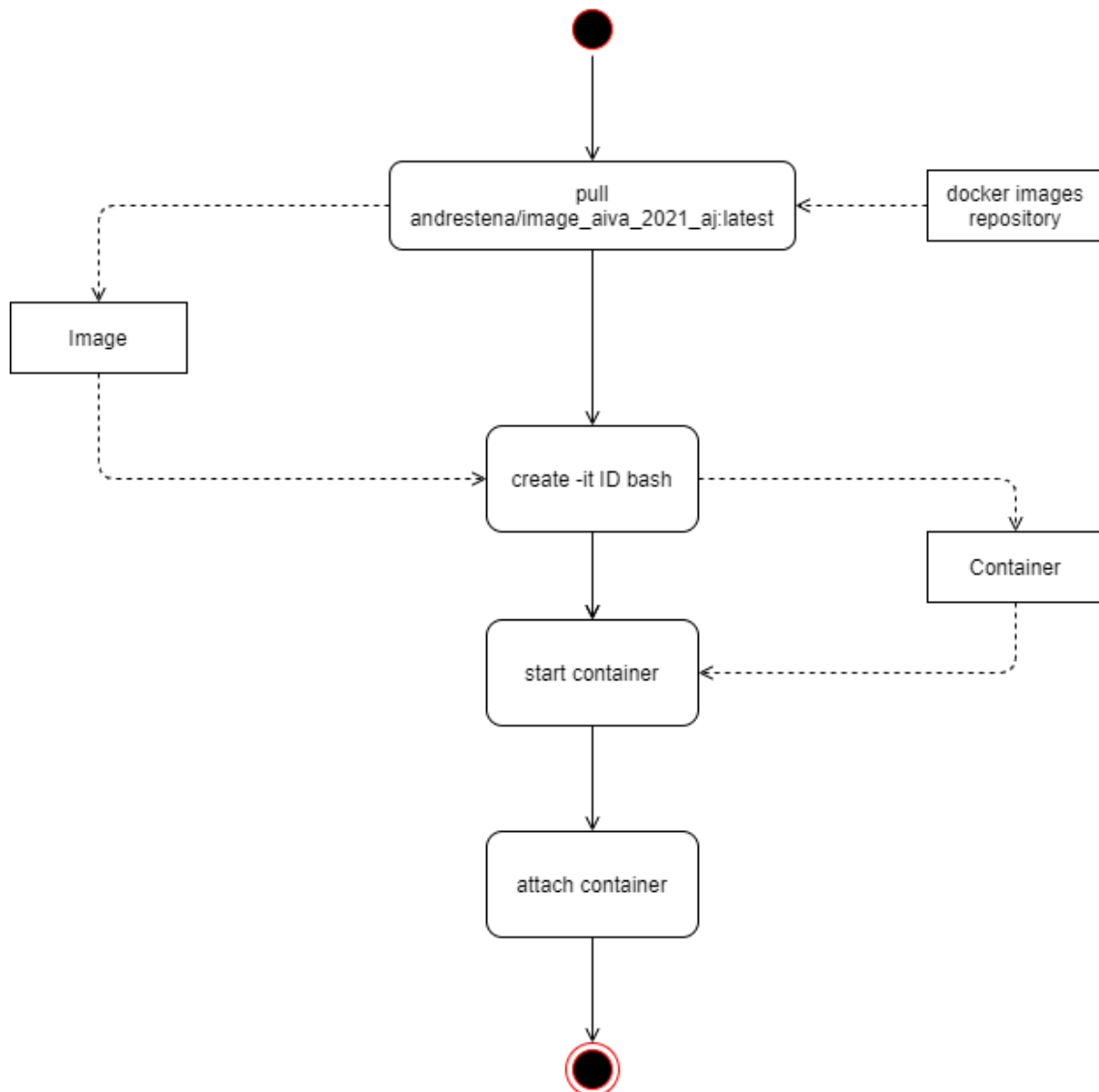
C:\Users\andre>docker start e4f2cd95f178
e4f2cd95f178

C:\Users\andre>docker attach e4f2cd95f178
root@e4f2cd95f178:/opencv# java /tmp/AIVA_2021_AJ/Java/src/co/ortizol/Main.java
Do you want to analyze a Train or Test image?
Test
What is the index of the image you want to analyze?
8
Image 8.png, percentage of imperfections:0.017980457490265268
root@e4f2cd95f178:/opencv#

```

## II. Despliegue

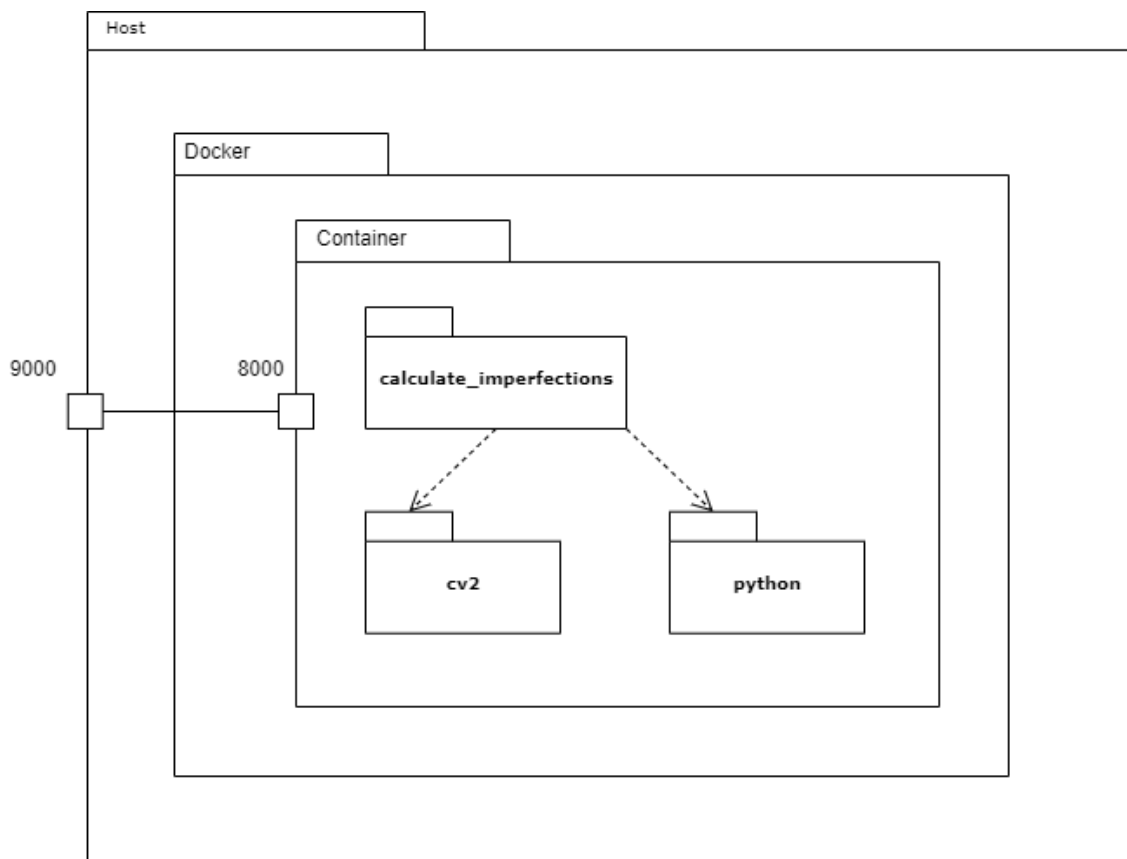
En esta sección se van a mostrar tanto el diagrama de despliegue de la aplicación en Docker y para ello será necesario mostrar como es el diagrama de despliegue de Docker.



*Figura 1: Despliegue docker.*

Visto como es el diagrama de despliegue de Docker, a continuación se pasará a mostrar el diagrama de la aplicación dentro de un contenedor Docker.





*Figura 2: Despliegue aplicación.*

Una vez se haya creado el contenedor de Docker que dispone de la aplicación, se podrá ejecutar la aplicación sin ningún tipo de problema ya que el contenedor también posee las dependencias que requiere el método *calculate\_imperfections*.

### III. Funcionamiento del sistema

La aplicación sigue el siguiente esquema, en primer lugar, se ejecuta un archivo Java el cuál pedirá la ruta de la imagen a analizar. Esta ruta será enviada al método *calculate\_imperfections* que leerá la imagen proporcionada utilizando la librería de opencv. A continuación, llamará al método *imperfection\_and\_segmentation(img)* de la clase *ImperfectionRecognizer* que devolverá una tupla con la máscara y el porcentaje de área con imperfecciones.

Para calcular dicha máscara *ImperfectionRecognizer* llama al método *do\_segmentation(hsv, background)* de *BackgroundSegmentator* que devolverá una máscara que representa el fondo de la imagen. De tal manera que discriminaremos así todo lo que no sea la pieza de madera.

Dicha máscara del fondo será necesaria para calcular la máscara de imperfecciones. Para ello *ImperfectionRecognizer* activa el método *do\_segmentation(hsv, background)* de la clase *ImperfectionsSegmentator*.

Una vez hemos calculado la máscara de imperfecciones, es necesario cuantificar que área abarcan dichas imperfecciones. Para esto *ImperfectionRecognizer* utilizará el método *do\_quantification(quantifiable, non\_quantifiable)* de la clase *MaskQuantificator*.

Finalmente, el porcentaje de imperfecciones de la imagen calculado será devuelto a Java para que este lo muestre por pantalla.

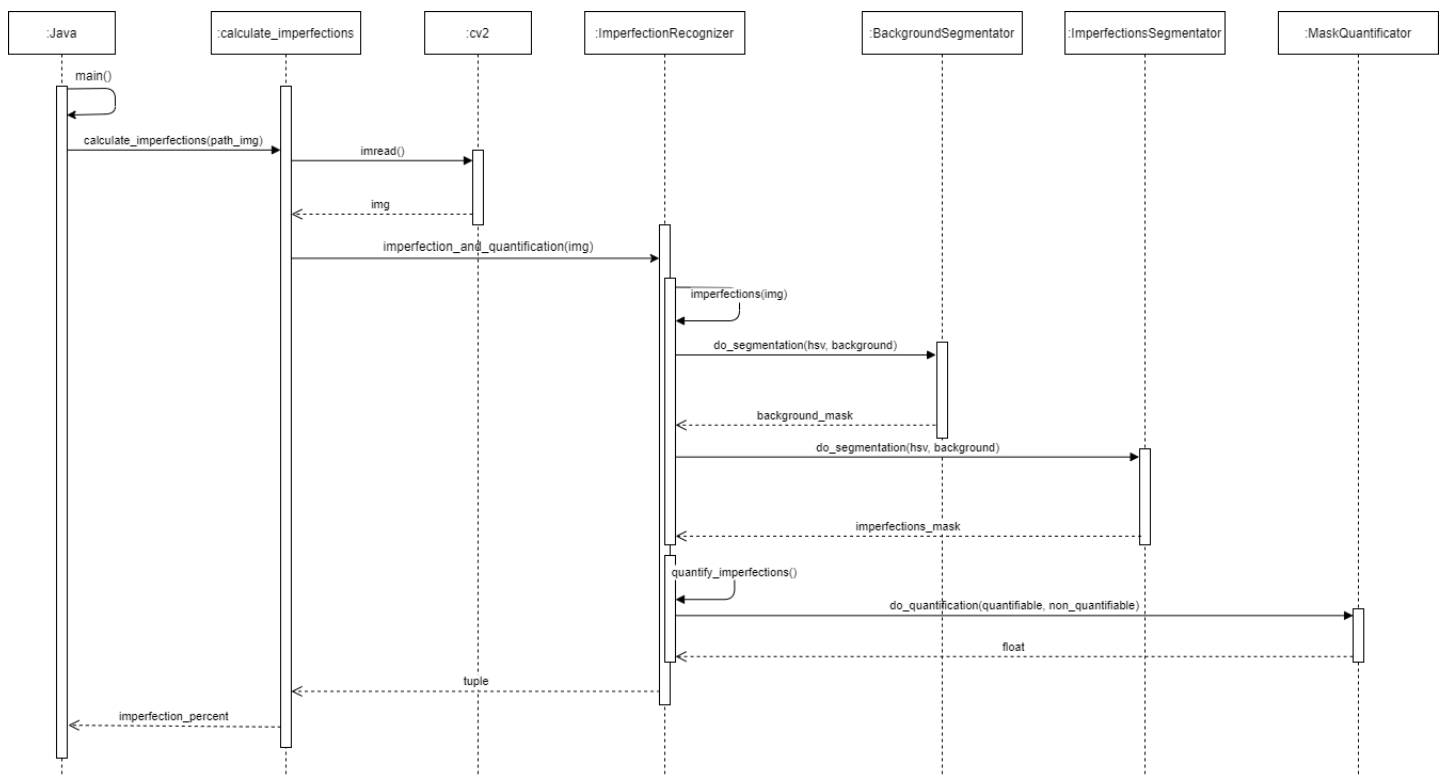


Figura 3: Diagrama de secuencia de la aplicación.