

Documentación Robot Template

Rocketbot necesita facilitar la revisión y aprobación en el menor tiempo posible de los BOTs enviados por los Partners para su publicación en el Marketplace. Se hace necesario estandarizar el proceso y la forma de construcción, por esta razón se ha habilitado el nuevo módulo ProbeForms en Rocketbot Studio, que les permitirá resolver la interfaz hacia el usuario y el ingreso de los datos requeridos para la operación del BOT. La idea es que los construyan usando este modulo de ingreso de datos y no desarrollar su propia solución, esto para agilizar la revisión y aprobación del mismo. Como una guía para ustedes se construyó este BOT con el fin de usarlo como plantilla para la creación de otros robots en MarketBot, proporcionando una estructura organizada para desarrollar robots especializados. Facilita la creación enfocados en dividir tareas específicas, como configuración, trabajo con Excel, envío de correos, etc. además de validación e ingreso de información a través de un formulario local, optimizando así el uso por parte del usuario. El proceso de ejemplo de este robot recorre y obtiene la información de un Excel para luego llenar un formulario web, obteniendo la respuesta y volviendo al Excel a indicar si fue procesado o no, finalmente envía un correo con el archivo adjunto.

INPUTS

VARIABLES MODIFICABLES

Robot	Variable	Valor Actual	Explicación
main	{espera_datos}	120	Son los 120 segundos de espera a que el usuario ingrese la información en el formulario, esta variable es utilizada en los comandos Wait submit del módulo ProbeForms

ESTRUCTURA

Proyecto –

- | - □docs (no usada)
- | - □logs
- | - □logs_bot
- | - □modules (no usada)
- | - □resources
- | - □robots (no usada)
- | - □screenshot (no usada)
- | - □main.bat
- | - □robot.db
- | - □README.md
- | - □package.json

MÓDULOS

Nombre	Versión
SystemAdvanced	2.8
ProbeForms	1.2.0
gmail_	1.12.7

ROBOTS

main

Robot principal, encargado de dejar todas las variables limpias antes de iniciar, posteriormente llama a los robots hijos para ejecutar cada parte del proceso.

config

Llama al robot para validar los procesos y valida si faltan datos para la configuración, si faltan o es la primera ejecución (todas las variables de config vacías) llamará al robot encargado del ingreso, sino, mostrará el formulario de reinicio para que el usuario decida si quiere reiniciar la configuración e ingresar de nuevo los datos o si prefiere continuar con los datos que ya se tienen.

termina_proceso

Valida si existen procesos de ProbeForms activos y los mata si corresponde.

ingresa_config

Muestra el formulario de ingreso de información y obtiene los datos ingresados por el usuario.

lee_excel

Abre el archivo Excel y extrae la información.

recorre_datos

Recorre los datos obtenidos del Excel y los guarda en sus variables correspondientes, dentro del ciclo llama al robot encargado de ir llenando el formulario web con estos datos, finalmente cierra el navegador y guarda los cambios en el Excel.

llena_form

Llena la información extraída del archivo Excel en el formulario web, valida

la respuesta entregada y escribe el status en la celda correspondiente.

envia_correo

Envía el correo a el/los destinatarios, adjuntando el archivo Excel.

escribe_log

Escribe la información enviada desde los robots en el archivo log customizado, además valida la información de la variable {detener}, si es True detendrá todo el proceso.

valida_res

Valida la respuesta de la variable {res}, si es False, cerrará el navegador y llamará al robot para escribir el log.

VIDEO EJECUCIÓN

¿CÓMO PROBARLO?

1- Descarga el siguiente .zip [\[link\]](#) y descomprimelo en la carpeta que desees

[template_marketbotDownload](#)

2- Descomprime el siguiente .zip [\[link\]](#) que contiene los módulos a utilizar

[mods-templateDownload](#)

3- Las 3 carpetas de los módulos (gmail_, ProbeForms y SystemAdvanced) pégalos en tu_carpeta_rocketbot/modules

4- Abre Rocketbot Studio y carga la base de datos robot.db que está en la raíz de la carpeta template_marketbot

5- Se mostrará una lista de robots, clickea en el que se llama main

6- Ejecuta el robot con el botón verde → Ejecutar

7- Te mostrará el formulario inicial donde deberás ingresar tu información

□ Considerar que los datos se deben ingresar en un tiempo máximo de 2 minutos, si no se ingresan dentro de ese rango, verás una ventana de que el robot se detuvo y esto significa que finalizó con error. Este tiempo está configurado por defecto, si necesitas aumentarlo, debes modificar la variable {espera_datos} en el robot main

Datos de Configuración

URL Formulario

Ruta de Excel

Carpeta para Logs

Correo:

Contraseña

Destinatarios

URL Formulario: Debe ser <https://roc.myrb.io/dev/forms/INGKKJ0HGGX0JNRZ>

Ruta de Excel: debes buscar la ruta de tu carpeta
template_marketbot/resources y elegir el archivo datos_form.xlsx

Carpeta para Logs: debes buscar la ruta de tu carpeta
template_marketbot/logs_bot

Correo: Debes ingresar el correo desde donde se enviará la información, debe ser un **gmail**

Contraseña: Debes ingresar la [contraseña de aplicación](#) de la cuenta gmail

Destinatarios: Debes ingresar el correo del destinatario, si es más de uno los debes separar por coma

8- Una vez ingresada el robot continuará con la ejecución

9- Una vez finalizada la ejecución clicar el botón Guardar, realizar esto sólo en la primera ejecución.

□ Con esto en la próxima ejecución preguntará si se quieren reiniciar o no

los datos de configuración, donde si se elige la opción SI, mostrará un formulario para volver a ingresarlos, en caso contrario el robot continuará con los datos previamente ingresados, ésto para evitar estar escribiendo lo mismo en cada ejecución y sólo realizarlo en caso de que alguna información cambie.

Usando XPath para Elementos Dinámicos

Cuando tratamos con elementos dinámicos en una página web, como aquellos cuyos identificadores XPath o Full XPath pueden cambiar, XPath ofrece una flexibilidad que resulta valiosa. Puedes crear expresiones que se centren en atributos específicos para asegurar la identificación precisa de los elementos, incluso si otros aspectos de la estructura de la página cambian.

Estrategias XPath para Elementos Dinámicos

Identificación por Atributo

Especificar un atributo específico de un elemento, como una clase, ID o cualquier otro atributo único.

```
//tagname[@class='claseEspecifica']
```

Esto asegura que, aunque la posición del elemento o la estructura de la página cambien, aún puedes dirigirte al elemento deseado basándote en un atributo constante.

XPath Relativo

Construir rutas XPath que son relativas al contexto del elemento, evitando dependencias absolutas que podrían volverse obsoletas con cambios en la estructura

```
//div[@id='contenedor']/p[@class='miClase']
```

Utilizando rutas relativas, tu XPath se adapta mejor a cambios en la jerarquía de la página.

Funciones de XPath

Utilizar funciones XPath para seleccionar elementos basados en contenido, posición, u otras características dinámicas

```
//div[contains(text(), 'TextoDinamico')]
```

Las funciones XPath, como `contains()`, permiten seleccionar elementos basados en el contenido variable que pueden tener.

Ventajas de Utilizar XPath para Elementos Dinámicos

- **Adaptabilidad:** XPath permite construir expresiones que se adaptan a cambios en la estructura de la página.
- **Precisión:** Al basarse en atributos específicos, XPath proporciona una identificación más precisa de los elementos.
- **Mantenibilidad:** Estrategias como XPath relativo facilitan la adaptación del scraper a modificaciones en la jerarquía de la página.

En resumen, al emplear XPath para elementos dinámicos y centrarte en atributos específicos, puedes construir expresiones más robustas y resistentes a cambios en la estructura de la página, asegurando una extracción de datos más fiable y duradera.

Tipos de Identificadores en Web Scraping

En este documento revisaremos algunos identificadores esenciales en el web scraping, tanto su significado, como ventajas y desventajas.

ID (Identificador Único)

- **Significado:** Cada elemento en una página web puede tener un atributo ID único. Es un identificador exclusivo asignado al elemento, permitiendo su identificación sin ambigüedad.
- **Ejemplo:** `<div id="miElementoUnico">Contenido</div>`
- **Ventajas:** Rápida identificación y acceso directo al elemento. Ideal cuando se necesita seleccionar un elemento específico.
- **Desventajas:** No todos los elementos tienen un ID único. Los IDs pueden cambiar, afectando la estabilidad.

Clase (Class)

- **Significado:** Los elementos pueden tener uno o más atributos de clase. Múltiples elementos pueden compartir la misma clase.
- **Ejemplo:** `<p class="miClase">Texto</p>`
- **Ventajas:** Útil para seleccionar grupos de elementos similares
- **Desventajas:** No garantiza la unicidad. Puede seleccionar más elementos de los deseados.

Selector de Etiqueta (Tag Selector)

- **Significado:** Se refiere al nombre de la etiqueta HTML (por ejemplo, `div`, `p`, `a`). Selecciona todos los elementos de esa etiqueta en la página.
- **Ejemplo:** `<h2>Título</h2>`

- **Ventajas:** Útil para seleccionar todos los elementos de un tipo específico en una página.
- **Desventajas:** Puede seleccionar demasiados elementos si la etiqueta es común en la página.

XPath (Ruta de Acceso XML)

- **Significado:** Es una expresión que define la ubicación de un elemento en un documento XML o HTML. Permite seleccionar nodos de manera precisa.
- **Ejemplo:** `//div[@id='miElementoUnico']`
- **Ventajas:** Proporciona flexibilidad y precisión. Es útil en situaciones donde otros identificadores no son suficientes.
- **Desventajas:** Puede ser largo y vulnerable a cambios en la estructura de la página.

Full XPath (Ruta de Acceso XML Completa)

- **Significado:** Especifica la ruta completa desde el nodo raíz hasta el elemento deseado.
- **Ejemplo:** `/html/body/div[1]/p[2]`
- **Ventajas:** Especificidad máxima en la selección.
- **Desventajas:** Altamente susceptible a cambios estructurales. Menos legible y mantenible.

Cuando los elementos son dinámicos podemos crear un Xpath con un atributo específico para hacerlo más robusto, más información acá [XPath para elementos dinámicos](#)

Name

- **Significado:** Se refiere al atributo “name” de un elemento HTML.
- **Ejemplo:** `<input type="text" name="usuario">`
- **Ventajas:** Útil en formularios y elementos interactivos.
- **Desventajas:** No todos los elementos tienen un atributo “name”. Puede no ser único.

Elección del Identificador:

La elección del identificador dependerá del contexto y los requisitos específicos del proyecto. Algunos aspectos a considerar:

- **Unicidad:** Si hay un identificador único disponible (ID), es preferible, ya que facilita la selección precisa del elemento.
- **Flexibilidad:** Las clases permiten seleccionar grupos de elementos similares, lo que puede ser útil si se requiere la extracción de múltiples elementos.

- **Estabilidad:** Si los identificadores cambian con frecuencia, es preferible utilizar estrategias más estables, como selectores de etiquetas o XPath.
- **Eficiencia:** Seleccionar identificadores más específicos puede mejorar la eficiencia del scraper al reducir la cantidad de datos a procesar.

En resumen, la elección del identificador dependerá de la situación específica, y es recomendable adaptar la estrategia de selección a las necesidades del proyecto.

Cómo obtener los identificadores web

Para poder controlar las páginas web, necesitamos acceder a los identificadores de los elementos para así especificar en cuál de ellos queremos realizar acciones. En este documento veremos el paso a paso para lograrlo.

Tomaremos como ejemplo la página [RPA Challenge](#)

Inspeccionar la página

Primero debemos inspeccionar la web para visualizar el código fuente, para esto podemos dar click derecho en cualquier lugar y seleccionar la opción **Inspeccionar**, también podemos utilizar los atajos de teclado CTRL + SHIFT + C o F12 para Windows o Command + SHIFT + C para macOS.

Veremos una ventana como esta

 *Consola del navegador*

Clickeamos la flecha que se encuentra en la esquina superior izquierda y una vez que esté de color azul, significa que ya podemos marcar cualquier elemento de la web. Seleccionamos el elemento deseado, y automáticamente se resaltará en azul en el código, abarcando todo el contenido del elemento elegido.

 *Seleccionando input*

Los elementos pueden tener diferentes tipos de identificadores (id, class, name, etc.), para más detalles, revisa la siguiente entrada [Tipos de Identificadores Web.](#)

Elemento Dinámico

Podemos ver que el elemento marcado tiene **id**, **name** y **class**

```
<input _ngcontent-c2="" ng-reflect-name="labelCompanyName" id="jnPCa"
```

```
name="jnPCa" class="ng-untouched ng-pristine ng-invalid">
```

Algo bastante común es que algunos de éstos sean dinámicos. Si recargamos la página y revisamos nuevamente, veremos que, tanto el id como el name cambiaron

```
<input _ngcontent-c2="" ng-reflect-name="labelCompanyName" id="UDONR"
name="UDONR" class="ng-untouched ng-pristine ng-invalid">
```

Esto sucederá cada vez que la página se recargue o se vuelva a abrir, por lo que nuestro robot fallaría si indicamos uno de estos datos, el **Xpath** tampoco nos serviría ya que como el id es dinámico, también sería diferente

```
//*[@id="jnPCa"]
```

```
//*[@id="UDONR"]
```

El **fullXpath** tampoco serviría, ya que si nos fijamos, los elementos cambian de posición

 *Posición inicial del elemento*

 *Posición del elemento cambiada*

Una forma óptima de controlar esto, es crear un Xpath centrándose en algún atributo específico, más detalles acá: [Xpath para elementos dinámicos](#).

Si nos fijamos en el elemento, veremos que lo único que no cambia es el siguiente atributo

```
ng-reflect-name="labelCompanyName"
```

Podemos entonces crear un Xpath utilizando esa información, de esta forma

```
//input[@ng-reflect-name="labelCompanyName"]
```

Para saber si está correcto, podemos probarlo en el mismo navegador, donde inicialmente abrimos la consola, seguimos en **Elements** y presionamos CTRL + F si estamos en Windows o Command + F si estamos macOS, esto habilitará la casilla de búsqueda (1) y pegaremos el xpath previamente mencionado, al hacerlo nos marcará el elemento tanto en el código (2) como en la vista web (3)

 (1) *Input de búsqueda*

(2) *Elemento marcado en código*

(3) *Elemento marcado en la vista web*

Si está todo correcto, entonces ese identificador es el que debemos utilizar en los comandos de Rocketbot para indicar el elemento con el que queremos trabajar.

Elemento No Dinámico

Ahora veremos el caso de un elemento donde su posición e identificadores no cambian, en este ejemplo tomaremos el botón para descargar el archivo Excel

 Botón descarga

El elemento es el siguiente, si recargamos y lo volvemos a revisar, seguirá teniendo la misma información

```
<a _ngcontent-cl="" class=" col s12 m12 l12 btn waves-effect waves-light
uiColorPrimary center" target="_blank"
href="./assets/downloadFiles/challenge.xlsx"> Download Excel <i _ngcontent-
cl="" class="material-icons right">cloud_download</i></a>
```

Obtenemos el fullXpath y lo utilizamos en el comando web de Rocketbot donde necesitamos indicar el elemento

```
/html/body/app-root/div[2]/app-rpa1/div/div[1]/div[6]/a
```

Cómo importar un Robot

Cuando nos comparten un archivo .bd o un .json o cuando queremos mover un robot de un ambiente a otro, debemos cargarlo a Rocketbot Studio, para eso debemos realizar lo siguiente:

Cargar BD

Nos vamos a la pantalla de Inicio y clickeamos el botón **Cargar DB de Proyecto**, esto nos abrirá el explorador, buscamos el archivo y lo cargamos.



Nos mostrará una lista con todos los robots dentro de esa Base de datos, seleccionamos el robot que necesitamos y ésto nos redirigirá al Dashboard donde visualizaremos el flujo.



Cargar Json

Nos vamos a la pantalla de Inicio y clickeamos el botón **Robot Nuevo**, ingresamos un nombre y lo creamos.



Una vez en el Dashboard podemos arrastrar nuestro archivo .json o clickear donde aparece el mensaje **“Arrastre aquí un archivo de Rocketbot Studio (json)”**

para editar", esto nos abrirá el explorador, lo buscamos y cargamos.

Nos mostrará un mensaje indicando que el nombre del robot actual cambiará, si clickeamos **OK**, dejará el nombre del robot que se está cargando, si clickeamos **Cancel**, conservará el nombre que indicamos al crear el robot nuevo.



Cómo exportar un Robot

Cuando tenemos un proyecto y queremos exportarlo para compartirlo o llevarlo al ambiente productivo, debemos realizar lo siguiente:

Ir a la pestaña **Robot** donde tendremos 2 opciones:

1. **Exportar robots a DB:** Nos mostrará un pop-up donde podremos elegir si queremos que nos descargue un **.zip** con los módulos utilizados en nuestro robot y si queremos exportar la Base de datos a Producción.



Si tenemos robot padre y robot hijos, para exportar todo el proyecto siempre debemos realizarlo desde el robot principal

- **Check seleccionado en "Incluir módulos":** Descargará un .zip con la carpeta modules, donde dentro se encontrarán todos los módulos que contiene el robot o proyecto.
- **Check seleccionado en "Exportar robots a producción":** Base de datos encriptada, no permite volver a cargar a Rocketbot Studio ni ver y/o modificar sus comandos. Sólo permite la ejecución del robot.
- **Check no seleccionado en "Exportar robots a producción":** Base de datos normal que se puede volver a cargar en Rocketbot Studio, ver su información y modificar. Permite descargar el Proyecto completo (varios bots que corresponden a un mismo proceso, robot padre e hijos).

2. **Guardar en archivo Json:** Sólo descarga un Robot, el que se tiene cargado actualmente en Rocketbot Studio.

Para cargar estos archivos a Rocketbot Studio, puedes revisar la siguiente entrada:

[Cómo importar un Robot](#)

Instalación Rocketbot Studio

Descubre en esta guía todos los pasos necesarios para llevar a cabo la instalación de Rocketbot Studio de manera sencilla y eficiente. Desde la descarga hasta la resolución de posibles alertas durante la instalación, te proporcionamos una completa orientación para que puedas comenzar a automatizar tus procesos de manera exitosa.

Windows

Requisitos Previos

Mínimos

Procesador Intel Core i3-4340 or AMD FX-6300
Almacenamiento 10GB
Memoria 4GB RAM

Recomendados

Procesador Intel Core i5
Almacenamiento 175GB
Memoria 8GB RAM

Descarga

1. Ir la página de Rocketbot [\[1\] Descargar Studio](#), al final de la página debemos clicar el botón Descargar para Windows, fijarse que sea la última versión.



2. Una vez descargado, descomprimos el archivo .zip en la ubicación que queramos
3. Abrir como **Administrador** el ejecutable llamado rocketbot.exe

Licenciamiento

Veremos 2 opciones, selecciona el tipo de licenciamiento



- **Get License:** Si no cuentas previamente con una licencia de Orquestador,

entonces elige esta opción para solicitar la licencia exclusiva de Rocketbot Studio.

- **Login:** Si tu administrador del Orquestador te proporcionó credenciales y token, entonces elige esta opción y úsalos para iniciar Rocketbot Studio.

Get License

Abrirá Rocketbot Studio en nuestro navegador predeterminado con la dirección **localhost:5000** y nos indicará que no contamos con una licencia activa, damos click en el botón Close



Nos mostrará un token el cual debemos copiar con el botón rojo ubicado a la derecha



Clickeamos en la palabra [aquí](#) la cual nos redirigirá a la página de licencias.



Licencia de Desarrollo:

Debemos completar los datos solicitados y en **Ambiente** debe estar marcado Desarrollo.

Licencia de Producción:

Debemos completar los datos solicitados y en **Ambiente** debe estar marcado Producción, esto habilitará 2 opciones más:

1. Orden de compra: Donde debemos adjuntar el PDF con la OC
2. CRM Code: Código entregado por tu PAM

Finalmente en la casilla de **Token** pegaremos el token anteriormente copiado desde Rocketbot Studio y enviamos el formulario.

A la casilla de correo indicada en el formulario, nos llegará la licencia



La copiamos, volvemos a Rocketbot Studio y la pegamos, finalmente damos click en el botón Validar



Nos mostrará un mensaje en la esquina inferior izquierda indicando que el licenciamiento fue exitoso.



Login

Nos mostrará una ventana para introducir las credenciales entregadas por nuestro administrador de Orquestador



Una vez ingresadas, si el login es correcto, nos mostrará el mensaje en la consola de Rocketbot y levantará Studio en nuestro navegador por defecto.



macOS

Requisitos Previos

1. Debemos tener instalado xcode, si no lo tenemos, lo podemos realizar con la siguiente instrucción desde el Terminal `xcode-select --install`
2. Debemos habilitar el Terminal en Privacidad y Seguridad
 - En versiones anteriores a Ventura, ir a **Seguridad y Privacidad → Privacidad → Herramientas para desarrolladores**



- Desde Ventura en adelante ir a **Seguridad y Privacidad → Herramientas para desarrolladores**



Descarga

1. Ir la página de Rocketbot [Descargar Studio](#), al final de la página debemos clicar el botón Descargar para macOS, fijarse que sea la última versión.
2. Una vez descargado, descomprimos el archivo .zip en la ubicación que queramos
3. Abrir la carpeta y dar doble click en el archivo ejecutable llamado **rocketbot**



4. Al dar doble click en el ejecutable, nos mostrará un mensaje de aviso donde indica que no se puede abrir la aplicación ya que no es un desarrollador identificado, damos click en OK para cerrar el aviso y realizar los siguientes pasos



Ir a **Preferencias del Sistema → Seguridad y Privacidad**



Nos aparecerá el siguiente mensaje, damos click en “Abrir”.



Esto nos abrirá nuestro navegador predeterminado con la dirección **localhost:5000**, damos click en Close



Licenciamiento

Veremos 2 opciones, selecciona el tipo de licenciamiento



- **Get License:** Si no cuentas previamente con una licencia de Orquestador, entonces elige esta opción para solicitar la licencia exclusiva de Rocketbot Studio.
- **Login:** Si tu administrador del Orquestador te proporcionó credenciales y token, entonces elige esta opción y úsalos para iniciar Rocketbot Studio.

Get License

Abrirá Rocketbot Studio en nuestro navegador predeterminado con la dirección **localhost:5000** (o el puerto que hayamos configurado en *rocketbot.ini*) y nos indicará que no contamos con una licencia activa, damos click en el botón Close



Nos mostrará un token el cual debemos copiar con el botón rojo ubicado a la derecha



Clickeamos en la palabra [aquí](#) la cual nos redirigirá a la página de licencias.



Licencia de Desarrollo:

Debemos completar los datos solicitados y en **Ambiente** debe estar marcado Desarrollo.

Licencia de Producción:

Debemos completar los datos solicitados y en **Ambiente** debe estar marcado Producción, esto habilitará 2 opciones más:

1. Orden de compra: Donde debemos adjuntar el PD con la OC

2. CRM Code: Código entregado por tu PAM

Finalmente en la casilla de **Token** pegaremos el token anteriormente copiado desde Rocketbot Studio y enviamos el formulario.

A la casilla de correo indicada en el formulario, nos llegará la licencia



La copiamos, volvemos a Rocketbot Studio y la pegamos, finalmente damos click en el botón Validar



Nos mostrará un mensaje en la esquina inferior izquierda indicando que el licenciamiento fue exitoso.



Login

Nos mostrará una ventana para introducir las credenciales entregadas por nuestro administrador de Orquestador



Una vez ingresadas, si el login es correcto, nos mostrará el mensaje en la consola de Rocketbot y levantará Studio en nuestro navegador por defecto.



Errores comunes

1. Si cuando ejecutamos Rocketbot nos arroja alertas de Python o de alguna librería como estos ejemplos:



Debemos cancelar la alerta y dar los permisos de seguridad correspondientes, ir a [Activar el Terminal](#)

2. Si no tenemos la opción de Herramientas para desarrolladores (Developer Tools), debemos abrir una ventana de Terminal y escribir el siguiente comando `spctl developer-mode enable-terminal` con eso ya nos debe aparecer.
3. Si al abrir **localhost:5000** nos aparece un error de acceso denegado a localhost, debemos cambiar el puerto de Rocketbot



Para cambiar el puerto debemos ir a la carpeta de Rocketbot y abrir el archivo `rocketbot.ini`



❏Nota:

Al cambiar el puerto debemos reiniciar rocketbot para que se vuelva a abrir con el nuevo

Server con Múltiples Usuarios

La duda recurrente es “¿Cómo es el licenciamiento en una máquina donde acceden múltiples usuarios al mismo tiempo?”.

Para este caso tenemos 3 opciones.

Opción 1:

Primero debemos revisar si dejando **Rocketbot Studio** en la raíz, pueden acceder todos los usuarios y utilizarlo, si es así, entonces sería 1 Licencia con X cantidad de ejecuciones en paralelo (la cantidad según el número de usuarios que se conectarán)

Opción 2:

Si no permite acceder a cada usuario a **Rocketbot Studio** instalado en la raíz (Opción 1), entonces debe ser una licencia por cada ambiente de usuario.

Opción 3:

Con la licencia Online se puede conectar cada usuario en cualquier máquina con **Rocketbot Studio instalado**, siempre y cuando tenga disponible la cantidad de conexiones necesarias.

Instalación Rocketbot en macOS

1. Descargar Rocketbot Studio

Ingresa a la página de [Rocketbot](#), clickear el botón **Descargue Rocketbot** y elegir el sistema operativo macOS.



2. Descomprimir

Una vez descargado, descomprimos el archivo .zip, obtendremos una carpeta llamada Rocketbot.



3. Ejecutar Rocketbot

Luego de descomprimir podemos dejar la carpeta donde estimemos conveniente, en este ejemplo quedará en Desktop.

Abrir la carpeta y dar doble click en el archivo ejecutable llamado **“rocketbot”**



3. Conceder permisos

Al dar doble click en el ejecutable, nos mostrará un mensaje de aviso donde indica que no se puede abrir la aplicación ya que no es un desarrollador identificado.

Realizaremos los siguientes pasos:

- Damos click en **OK**



- Ir a **Preferencias del Sistema – Seguridad y Privacidad**

Nos aparecerá que el uso de rocketbot se bloqueó, debemos dar click en **“Abrir de todos modos”**.



Nos aparecerá el siguiente mensaje, damos click en **“Abrir”**.



- Esto nos abrirá nuestro navegador predeterminado con la dirección **localhost:5000**



Ya podemos solicitar la licencia en: <http://license.rocketbot.co/>

5. Consideraciones Extra

Si además de permitir que abra Rocketbot, nos arroja alertas de Python o de alguna librería como estos ejemplos:



Debemos cancelar la alerta y dar los permisos de seguridad correspondientes.

- En la ventana de **“Seguridad y Privacidad”**, ir a la pestaña **“Privacidad”**,

en el menú izquierdo hacer scroll y seleccionar **“Herramientas para desarrolladores”**.



Si no tenemos la opción de **Herramientas para desarrolladores** (Developer Tools), debemos abrir una ventana de Terminal y escribir el siguiente comando `spctl developer-mode enable-terminal` con eso ya nos debe aparecer

- También debemos instalar xcode con la siguiente instrucción desde el Terminal `xcode-select --install`

Nos aparecerá una ventana indicando que debemos instalar las tools, clickeamos **Install** y esperamos que termine



- Volvemos a Seguridad y Privacidad y clickeamos el candado de la esquina inferior izquierda para poder realizar cambios.



Esto nos pedirá la contraseña de nuestro usuario, la ingresamos y seleccionamos **“Desbloquear”**.



- Ahora podemos dar permisos al **Terminal**, en el caso que tengamos además otro terminal (ej: **iTerm**) y lo utilicemos para ejecutar rocketbot, también se debe seleccionar, luego seleccionamos el candado para guardar los cambios.



- Finalmente podemos ejecutar rocketbot nuevamente y ya nos abrirá **localhost:5000** en nuestro navegador predeterminado.

Si al abrir localhost:5000 nos aparece un error de acceso denegado a localhost, debemos cambiar el puerto de Rocketbot



Para cambiar el puerto debemos ir a la carpeta de Rocketbot y abrir el archivo `rocketbot.ini`



y cambiar donde dice port, elegimos otro puerto, por ejemplo el 5003



quedando de esta forma



Volvemos a ejecutar rocketbot y ahora ya nos cargará en el nuevo puerto pidiéndonos activar la licencia



Ejecutar en Paralelo

Rocketbot permite ejecuciones en paralelo, lo que quiere decir que podemos lanzar más de 1 robot al mismo tiempo.

Uno de los requisitos es contar con una licencia de Producción que permita este tipo de ejecuciones.

Al iniciar nuestro Rocketbot Studio, nos informará en la consola cuántos robots en paralelo nos permite:



Tener en cuenta que al estar Rocketbot Studio abierto es considerado un proceso, por lo tanto si tengo un máximo de 2 procesos en paralelo, y requiero ejecutar 2 robots, debo cerrar el Studio.

Para iniciar más de un robot, lo podemos realizar a través de un .bat si estamos en Windows o desde un .sh si estamos en macOS/Linux.

Windows

El archivo .bat debe tener lo siguiente:

```
cd C:/ruta/a/carpeta/Rocketbot
START rocketbot.exe -start=robot1
START rocketbot.exe -start=robot2
```

macOS/Linux

El archivo .sh debe tener lo siguiente:

```
cd /ruta/a/carpeta/Rocketbot
./rocketbot -start=robot1 &
./rocketbot -start=robot2
```

Luego podemos ejecutar el .bat/.sh mediante un doble click o agregarlo a una tarea programada.

Para saber que procesos se pueden ejecutar en paralelo revise este link [¿que considerar para ejecucion en paralelo?](#)