

Escuela Agrícola Panamericana Zamorano

GREENHOUSEWARE

Proyecto CAPSTONE

Integrantes

Julia Janeth Leiva Guerra (24189) - Coordinadora

Geraldine Enciso Huamani (24187)

Nadya Paula Granda Flores (24085)

Valeria María Calderón Sánchez (24261)

Estrella De María Rodríguez Zuniga (21305)

Raúl Francisco Quiroz Rivera (24288)

Yohel Alejandro Rivera Saucedo (24153)

Oscar Eduardo Sánchez Aguilar (23006)

Gerardo Alexis Sosa Casco (24280)

Profesor: Dr. Servio Palacios

Instructor del Laboratorio: Ing. Gonzalo Armando Maradiaga Solano

Equipo

Strong Technology

Proyecto

GREENHOUSEWARE Software especializado en parámetros para un invernadero

- ✓ ENLACE DEL REPOSITORIO CREADO EN GITHUB: <https://github.com/Strong-Tecnology>
- ✓ ENLACE DEL PROYECTO EN GITHUB:
<https://github.com/StrongTecnology/GREENHOUSEWARE>

05/11/2022

ÍNDICE

• Introducción.....	3
• Definición del problema.....	3
• Objetivos.....	4
• Solución propuesta.....	5
• Métodos.....	5-6
• Fuente de datos.....	6-7
• Desarrollo del proyecto	7-18
• Conclusiones	18
• Referencias.....	19

GREENHOUSEWARE

INTRODUCCIÓN

Nuestro proyecto se basará en el control de invernaderos, como sabemos los invernaderos centran su atención en propiciar las condiciones óptimas para el cultivo y mejorar sus rendimientos. Esto significa controlar y monitorear las condiciones ambientales óptimas para el cultivo seleccionado y sembrado en el interior de estos sistemas.

Nuestro proyecto consiste en la realización de un software especializado en monitorear y controlar múltiples parámetros que se requieren en un invernadero y que en la actualidad son parámetros que no son tan controlados diariamente y algunos no hacen estudios de factibilidad acerca de ellos. A continuación, se muestran las principales variables a monitorear y controlar dentro de los invernaderos.

Parámetros Para Evaluar:

- Número de unidades producidas
- Tiempo de cosecha
- Rendimiento de la producción

DEFINICIÓN DEL PROBLEMA A RESOLVER

Múltiples invernaderos no cuentan con muchos parámetros que evaluar o solamente cuentan con los que creen necesarios, generalmente y la mayoría solo controlan la temperatura y humedad relativa con dispositivos especializados y no toman en cuenta otros parámetros que son de bastante utilidad, como por ejemplo medir el tiempo en que se cosecha, número de unidades producidas, por último saber cuánto será el rendimiento total de cada producción, que este principal parámetro es uno de los más fundamentales que deberían de considerar ya que eso permitirá saber que tan viable y factible será la producción de cultivos que se ejecutara en ese momento para que exista un mejor control

y mejores resultados, disminuyendo los porcentajes de pérdida y que la producción alcance su éxito en el menor tiempo posible.

OBJETIVOS

¿Qué vamos a realizar y cómo?

1. Nuestro principal objetivo es crear un software especializado para que lleve el control de los siguientes parámetros:
 - Número de días establecidos para cosecha de unidades.
 - Número de unidades producidas por cada ciclo del cultivo
 - Rendimiento Total de las producciones
2. Considerando estos aspectos el software podrá brindar los datos de cada parámetro automáticamente cada vez que le solicitemos, con la intención de que todos estos parámetros los pueda contener el software y se lleve un control y monitoreo detallado constantemente de cada uno de ellos.
3. Permitir la facilidad de acceso en el software sobre los parámetros mencionados para que las personas empiecen a implementarlos, para llevar a cabo una producción exitosa desde su inversión hasta la finalización de la producción, así mismo que le permita adquirir el conocimiento necesario para que lo pueda utilizar adecuadamente, conocimientos sobre como evaluar los ciclos de cada cultivo cada día, monitorear el número de unidades producidas y medir el rendimiento total que obtendrá por su producción.
4. Manejar las herramientas de los sistemas de información que vamos a utilizar para poner en práctica nuestros conocimientos adquiridos en la clase y conocimientos de investigación para la realización del software.

5. Implementar un software capacitado que brinde los resultados de los análisis estadísticos por medio de tablas que permitan brindar la información de cada parámetro cada vez que le solicitemos.
6. La creación de un repositorio en Github para almacenar la información completa de todo nuestro proyecto para poder publicarlo.

SOLUCIÓN PROPUESTA

La solución que nosotros brindamos para que las personas tomen en cuenta estos parámetros fundamentales para la ejecución de un proyecto, es realizar un software capacitado que pueda monitorear y controlar los parámetros ya mencionados anteriormente esto con la finalidad de que la persona pueda obtener mejores resultados y agilizar los procesos de producción teniendo los datos de estos y que lleve un control si es posible diariamente, la adición de nuevos parámetros como monitoreo de cada día que pase, número de unidades producidas, y rendimiento total de la producción le ayudará a determinar qué tan factible o exitosa está siendo la producción que este llevando a cabo en ese momento, y sobre todo un mejor cuidado intensivo de los cultivos dentro del invernadero y generar el más mínimo porcentaje de perdidas. Podemos agregar también que esta será una herramienta muy útil y de tendencia, así como también de innovación.

MÉTODOS

El software llamado GREENHOUSEWARE lo realizaremos de la siguiente manera:

Utilizaremos una variedad de herramientas para la creación y desarrollo del software con las características que nosotros deseamos, las herramientas que utilizaremos son las siguientes:

- **Visual studio code:** Que permitirá la codificación correspondiente para el desarrollo del software ya que será nuestro editor de fuente y que servirá como extensión para el programa Python.

- **Github:** Este sistema nos ayudará a guardar toda la información que implementemos en Visual Studio Code, creando repositorios para almacenar todos los datos y requerimientos del proyecto en general.
- **Python:** Una vez ya completado los pasos mencionados con anterioridad a través de los sistemas de información que utilizaremos, finalmente procederemos a usar el programa Python para correr el software que estaremos desarrollando, ya que la funcionalidad de este programa consiste en crear programas, softwares, aplicaciones construir páginas web, analizar datos, automatizar operaciones.
- **Pandas:** Utilizamos esta herramienta como una librería para análisis y manipulación de datos obtenidos en las aplicaciones anteriores, contienen las estructuras de datos y operaciones necesarias para la elaboración de las tablas numéricas.
- **Jupyter:** Es una herramienta parecida a Pandas al desarrollar softwares de código abierto, pero con el objetivo de ser una plataforma interactiva para la configuración y organización de códigos, datos y cuadernos basado en la web.
- **Matplotlib:** Una biblioteca donde se crean visualizaciones estáticas, animadas o interactivas con el mismo lenguaje de Python, para plasmar y representar los datos de una manera más clara con el uso de gráficos de barra, entre otros.

FUENTE DE DATOS

En este proyecto nos enfocaremos en la realización del software de tres parámetros para invernaderos, el cual para nuestro primer parámetro que es el sistema de riego utilizaremos los archivos exportados de la empresa Galcon, un líder mundial en la fabricación de sistemas integrales de control de riego y Fertilización, dicho archivo se encuentra en a PDF con 137 páginas.

- Fuente de datos
[TEA-Data \(1\).xlsx](#)
- El URL de la fuente de datos (data source o dataset) se encuentra en el siguiente URL:

<https://www.galconc.com/wp-content/uploads/2020/06/Greenhouse-spanish.pdf>

- Para nuestro segundo parámetro utilizaremos la página web de scielo.org en el cual especifica una investigación completa con fuentes confiables sobre los ciclos del cultivo.

El URL de la fuente de datos (data source o dataset) se encuentra en el siguiente URL:

http://ve.scielo.org/scielo.php?script=sci_arttext&pid=S0378-78182007000300004

- Por último, nuestro tercer parámetro, rendimiento total de las producciones utilizaremos la página web oficial de InsightSoftware proveedor líder de soluciones de informes, análisis y gestión del rendimiento.

El URL de la fuente de datos (data source o dataset) se encuentra en el siguiente URL:

<https://insightsoftware.com/es/blog/30-manufacturing-kpis-and-metric-examples/>

Formato de la Fuente de Datos

A través de CSV Python que será nuestro formato de fuente codificaremos con algoritmos que permitan realizar y calcular los datos que ingresemos en el para obtener respuestas de los cálculos así mismo con CSV Python podremos obtener un tipo de documento que presente los datos que necesitaremos en forma de tabla.

DESARROLLO DEL PROYECTO

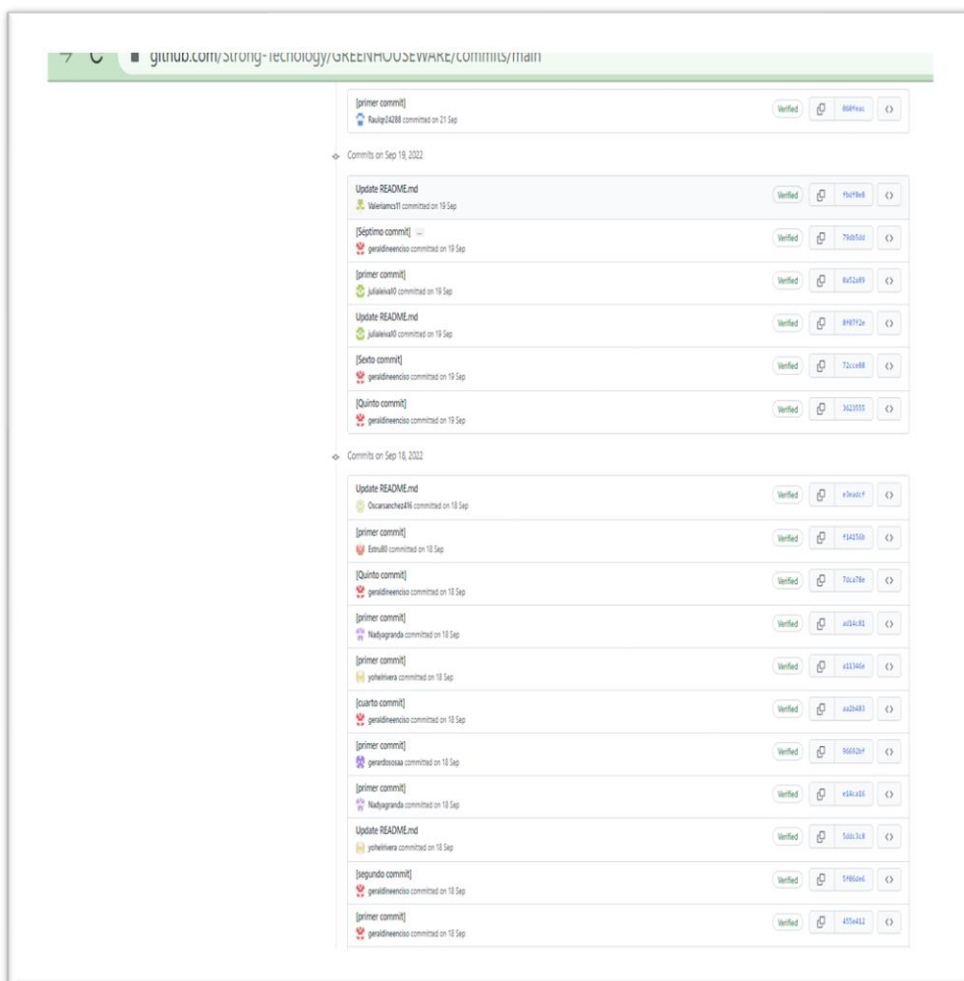
A lo largo del periodo académico se fue realizando diferentes actividades con respecto al avance del proyecto, fuimos analizando los datos del documento que está en el formato CSV titulado “TEA-Data.csv” que está subido a GitHub, asimismo, fuimos implementando algunos códigos para poder hacer un análisis descriptivo y exploratorio y después ir creando algunas gráficas que nos ayudaron a poder identificar los días en dónde más resalto el rendimiento de la producción y muchas cosas muy importantes a tomar en cuenta.

ENLACE DEL REPOSITORIO CREADO EN GITHUB: <https://github.com/StrongTechnology>

ENLACE DEL PROYECTO EN GITHUB:

<https://github.com/StrongTechnology/GREENHOUSEWARE>

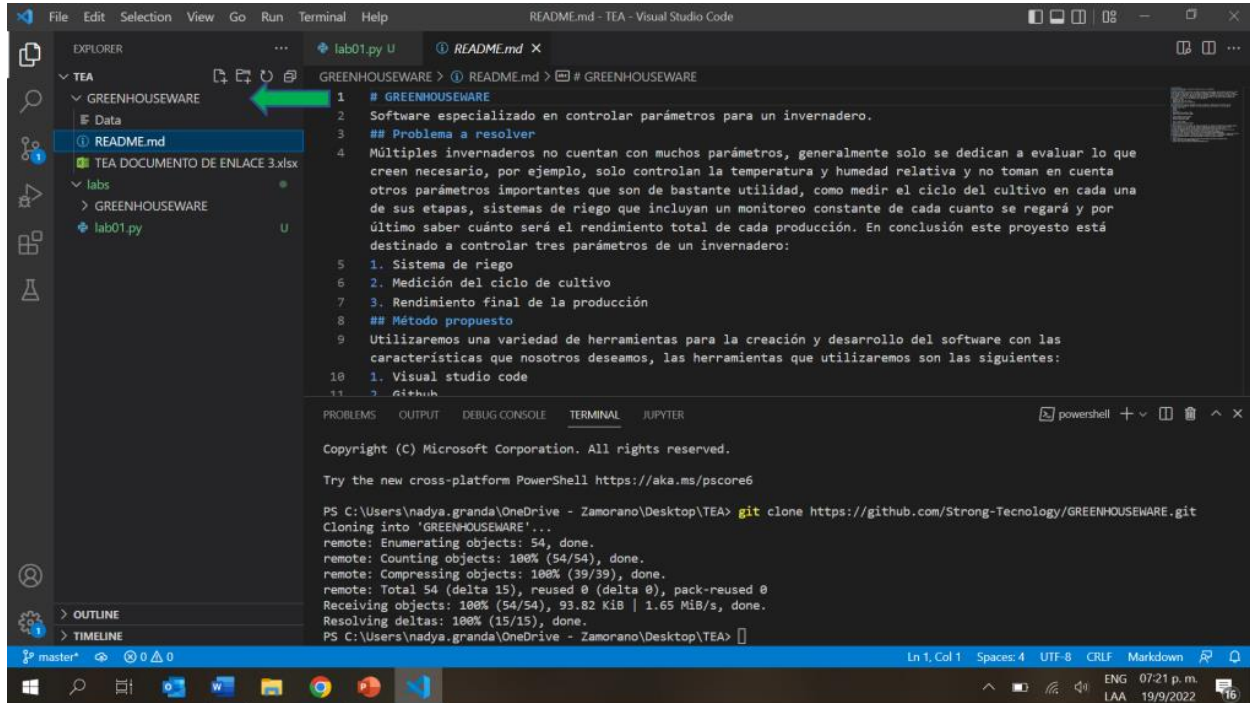
COMMITTS DE LOS INTEGRANTES DEL GRUPO



Laboratorio 3

Nuestro grupo empezó creando un repositorio en GitHub, titulado Strong Technology, además, esto se pudo enlazar a Visual Studio Code para que desde aquí podamos programar y subir a GitHub.

Clonación:



The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows a project structure with folders like 'TEA', 'GREENHOUSEWARE', and 'Data'. The README.md file is open in the editor, showing a document about 'GREENHOUSEWARE'. The terminal window at the bottom shows the execution of the command `git clone https://github.com/Strong-Tecnology/GREENHOUSEWARE.git`, which successfully clones the repository into the 'GREENHOUSEWARE' folder.

En esta captura de pantalla podemos observar la información de nuestro proyecto, en dónde cada integrante fue adicionando lo que le correspondía. Con esto empezamos a implementar nuestro proyecto de acuerdo con las clases recibidas.

Formato de la fuente de datos

CSV

Líneas de información

14 líneas con información en cada una de ellas.

Atributos de la fuente de datos (Ejemplo)

Product, Size, Color, Price

Shirt, Medium, Blue, \$14

Shirt, Large, Red, \$15

Pants, Medium, Khaki, \$23

Para el proyecto:

Tiempo en días, Número de unidades producidas, Rendimiento total de la producción

14 días, rendimiento en relación con el número de unidades producidas

Usaremos 3 datos para el proyecto:

- ✓ Número de días establecidos para cosecha de unidades.
- ✓ Número de unidades producidas por cada ciclo del cultivo
- ✓ Rendimiento Total de las producciones

Enlace del documento usado para el proyecto, cabe recalcar que de aquí se realizó una limpieza para solo obtener los datos numéricos. [https://github.com/Strong-Tecnology/GREENHOUSEWARE/blob/79db5ddc0dca4871713457a24104d837e04f040b/TEA%20DOCUMENTO%20DE%20ENLACE%203%20\(2\).csv](https://github.com/Strong-Tecnology/GREENHOUSEWARE/blob/79db5ddc0dca4871713457a24104d837e04f040b/TEA%20DOCUMENTO%20DE%20ENLACE%203%20(2).csv)

Información sobre gráficas

Podremos realizar y aplicar visualización gráfica, mediante gráficos de líneas y barras, etc. Además, usaremos promedios entre los datos que requerimos, sumas, restas para poder analizarlos de una manera óptima. Por otro lado, aplicaremos el razonamiento lógico para saber cuál sería la estimación del rendimiento total de la producción. El propósito de la creación de esta fuente de datos es específicamente para calcular los 3 datos generales para el proyecto, como el número de días, tiempo, número de unidades producidas y rendimiento total de producción. Los cuales se subdividen en otras propiedades y atributos para calcular la cantidad o cualquier otro dato que necesitaremos para los tres parámetros con los que vamos a trabajar. Esto con la finalidad de alcanzar los objetivos previamente mencionados en el documento del proyecto.

Descripción de las funciones y módulos utilizados.

1. **Import** = “Bring this functionality or library to my python script”
2. **Pandas** = The library you want to import, in this case, it is pandas
3. **As** = The python nomenclature for creating as alias. This is a fancy way of taking a long word and referencing it as a short word

4. **pd** = The standard short name for referencing pandas

Import - How you're telling python you want to bring another package/library in

As - Creating an alias. 'As' tells python that you're going to call "pandas" as "pd" from now on

```
import pandas as pd
```

pandas - The package/library you want to import. In this case, it's pandas

pd - The standard way to reference pandas. This means instead of typing "pandas.function" you'll only need to write "pd.function"

5. Usamos la convención estándar para hacer referencia a la API de matplotlib:

6. **En [1]: importar** matplotlib. **pyplot** como **plt**

7. Diagrama de barras: Para datos etiquetados que no sean series de tiempo, es posible que desee generar un gráfico de barras:

- En el proyecto usamos Llamar al método de trazado de un DataFrame con `kind='bar'` produce un gráfico de barras múltiples

8. Definir la figura que contendrá el gráfico, que es la región (ventana o página) donde se dibujará y los ejes sobre los que se dibujarán los datos. Para ello se utiliza la función `subplots ()`.

9. Para diagrama de barras: `plot (x, y)`: Dibuja un polígono con los vértices dados por las coordenadas de la lista `x` en el eje X y las coordenadas de la lista `y` en el eje Y.

CÓDIGOS FUENTES

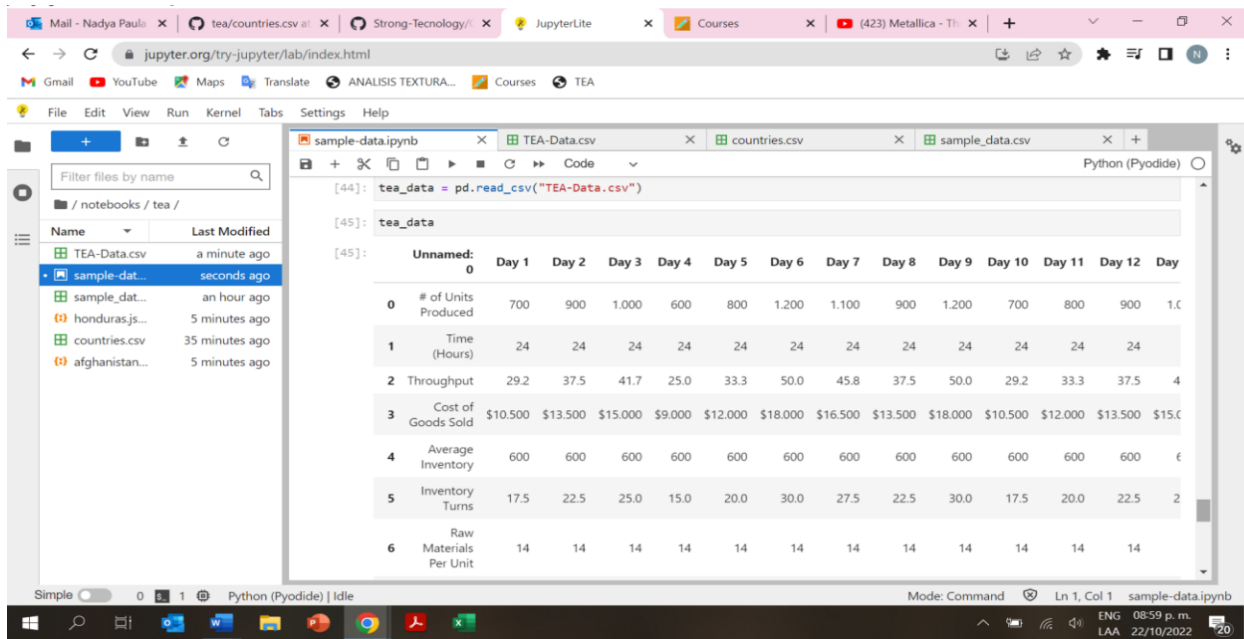
A lo largo del proyecto y en el repositorio de GitHub se irán presentando los códigos fuentes utilizados mediante capturas de pantallas, para esto mostraremos el paso a paso desde que inicio nuestro proyecto, mediante los laboratorios semanales.

Laboratorio 6

Durante este laboratorio fuimos investigando la fuente de datos, usamos pandas y matplotlib.

- Como ya sabemos **Pandas** es una librería de Python especializada en el manejo y análisis de estructuras de datos.
- Matplotlib es una biblioteca completa para poder crear visualizaciones estáticas, animadas e interactivas en Python.

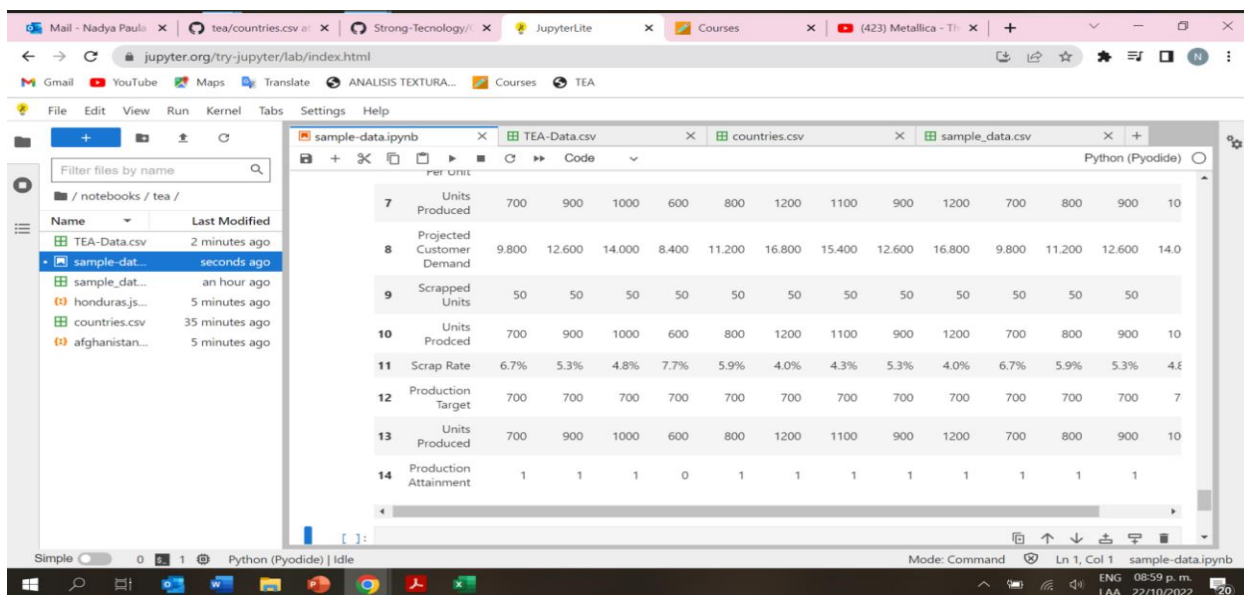
A continuación, mostramos la base de datos usada para el proyecto usando pandas:



The screenshot shows a JupyterLab interface with a file browser on the left and a code editor on the right. The code editor displays the following code and output:

```
[44]: tea_data = pd.read_csv("TEA-Data.csv")
[45]: tea_data
```

		Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day
0	# of Units Produced	700	900	1,000	600	800	1,200	1,100	900	1,200	700	800	900	1,000
1	Time (Hours)	24	24	24	24	24	24	24	24	24	24	24	24	24
2	Throughput	29.2	37.5	41.7	25.0	33.3	50.0	45.8	37.5	50.0	29.2	33.3	37.5	41.7
3	Cost of Goods Sold	\$10,500	\$13,500	\$15,000	\$9,000	\$12,000	\$18,000	\$16,500	\$13,500	\$18,000	\$10,500	\$12,000	\$13,500	\$15,000
4	Average Inventory	600	600	600	600	600	600	600	600	600	600	600	600	600
5	Inventory Turns	17.5	22.5	25.0	15.0	20.0	30.0	27.5	22.5	30.0	17.5	20.0	22.5	25.0
6	Raw Materials Per Unit	14	14	14	14	14	14	14	14	14	14	14	14	14



The screenshot shows a JupyterLab interface with a file browser on the left and a code editor on the right. The code editor displays the following code and output:

```
PER UNIT
```

		Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day
7	Units Produced	700	900	1000	600	800	1200	1100	900	1200	700	800	900	1000
8	Projected Customer Demand	9,800	12,600	14,000	8,400	11,200	16,800	15,400	12,600	16,800	9,800	11,200	12,600	14,000
9	Scrapped Units	50	50	50	50	50	50	50	50	50	50	50	50	50
10	Units Produced	700	900	1000	600	800	1200	1100	900	1200	700	800	900	1000
11	Scrap Rate	6.7%	5.3%	4.8%	7.7%	5.9%	4.0%	4.3%	5.3%	4.0%	6.7%	5.9%	5.3%	4.8%
12	Production Target	700	700	700	700	700	700	700	700	700	700	700	700	700
13	Units Produced	700	900	1000	600	800	1200	1100	900	1200	700	800	900	1000
14	Production Attainment	1	1	1	0	1	1	1	1	1	1	1	1	1

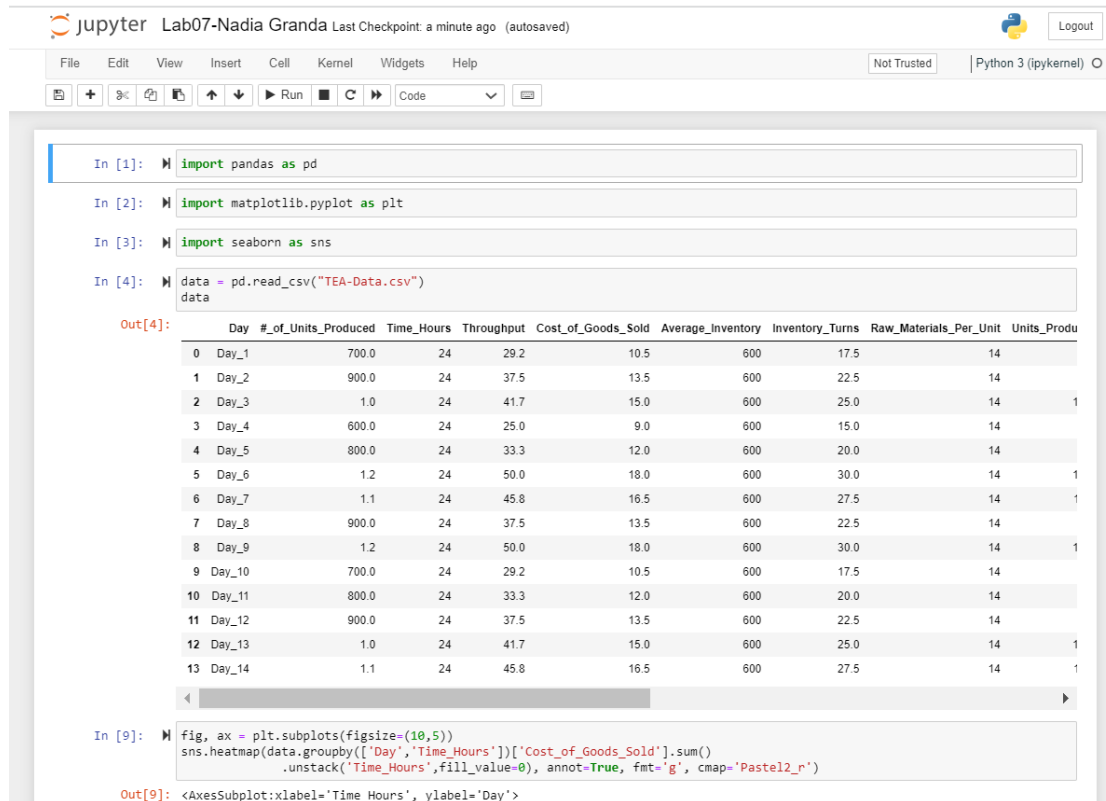
En las capturas se pueden observar el número de días, número de unidades producidas y el rendimiento de la producción, que son los parámetros que necesitamos para poner

en ejecución el proyecto. Esto lo hicimos con ayuda de Jupyter Notebook y lo subimos a GitHub en el repositorio mencionado anteriormente.

Laboratorio 7

Fuente de Datos del Proyecto

Analizando la Fuente de Datos del Proyecto

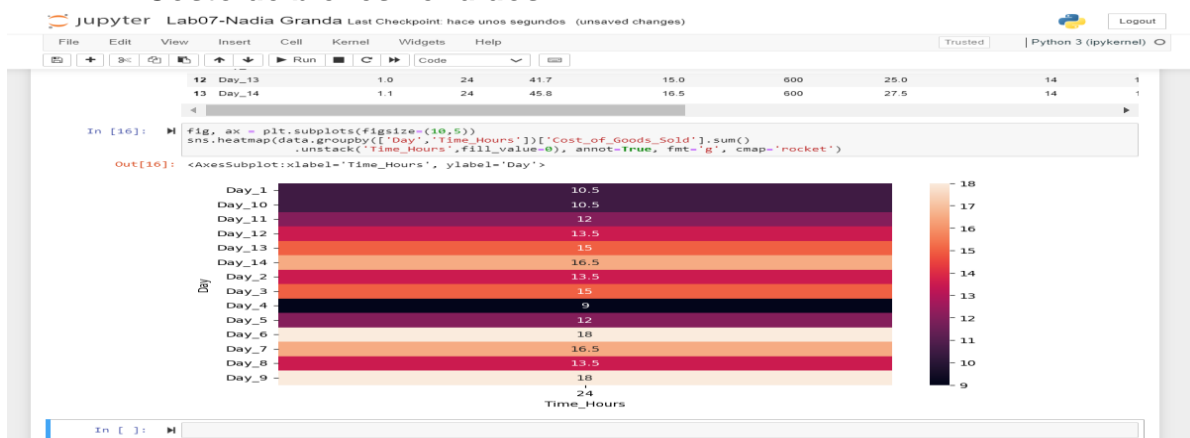


Se importó el documento con ayuda de pandas usando (import pandas as pd), gracias a esto pudimos obtener la base de datos para analizarla en Jupyter. Podemos observar diferentes datos en la tabla, lo que más destacaremos es el número de días, número de unidades producidas, tiempo en horas y el rendimiento del producto. Cabe recalcar, que se mejoró la estructura de la tabla que se realizó en el laboratorio número 6.

Resultados estadísticos del proyecto

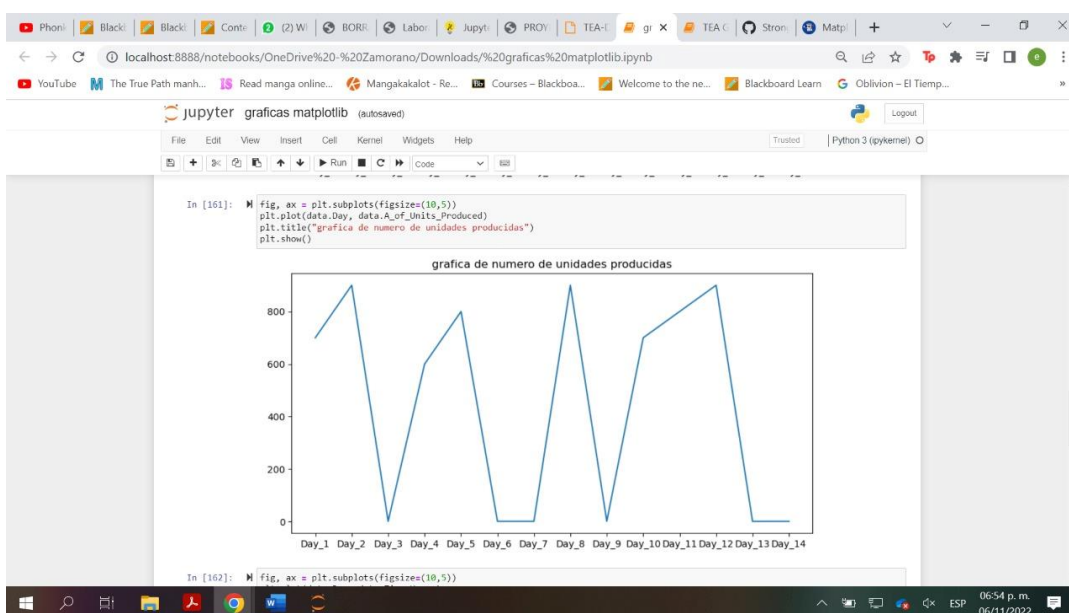
Con ayuda de diversos códigos puestos en ejecución pudimos realizar gráficos de líneas y barras para analizar la base de datos. A continuación, se explicará cada una:

1. Costo de bienes vendidos:



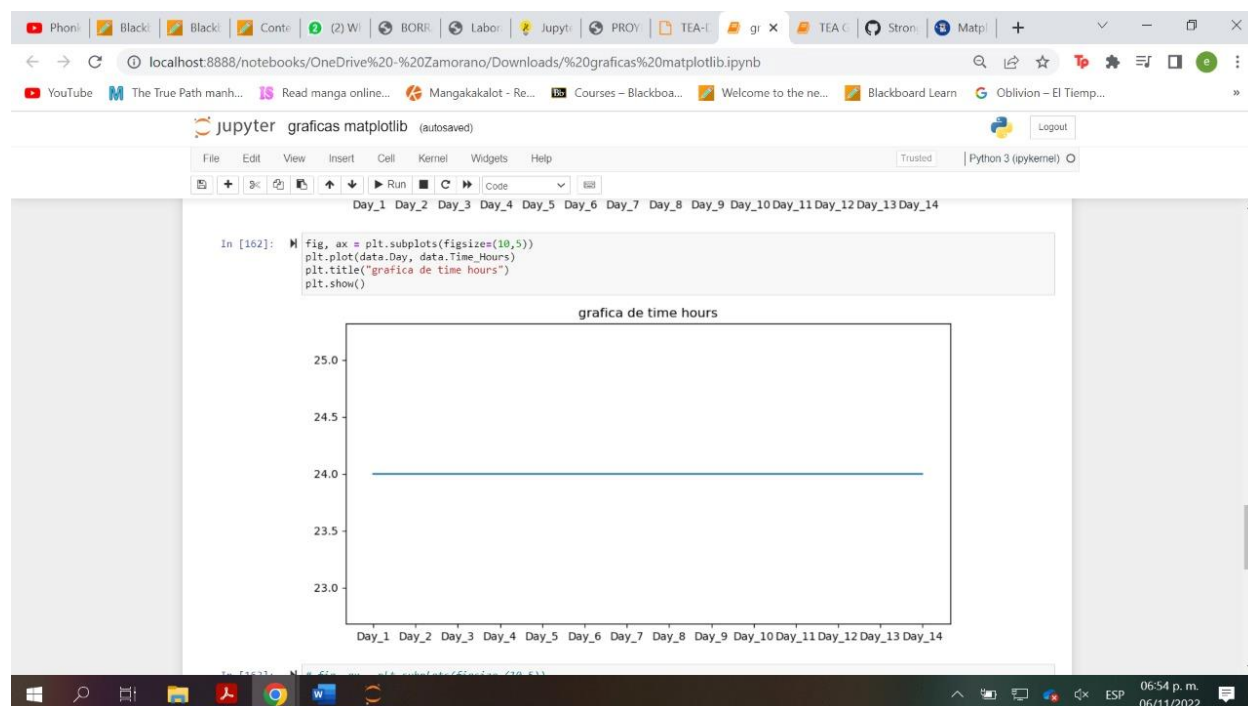
Gráfica N° 1, se puede ver el número de días que son 14, el tiempo en horas (24) y el Costo de los bienes vendidos mostrada en una escala de colores, en dónde el número más bajo tiene un color intenso y el más claro es el día en dónde obtuvimos mayores costos. Cabe recalcar que cada valor de los costos vendidos está en miles, se debe multiplicar por 1000 el número que está dentro de la barra del color y de esta manera sabremos el valor real.

2. Número de unidades producidas



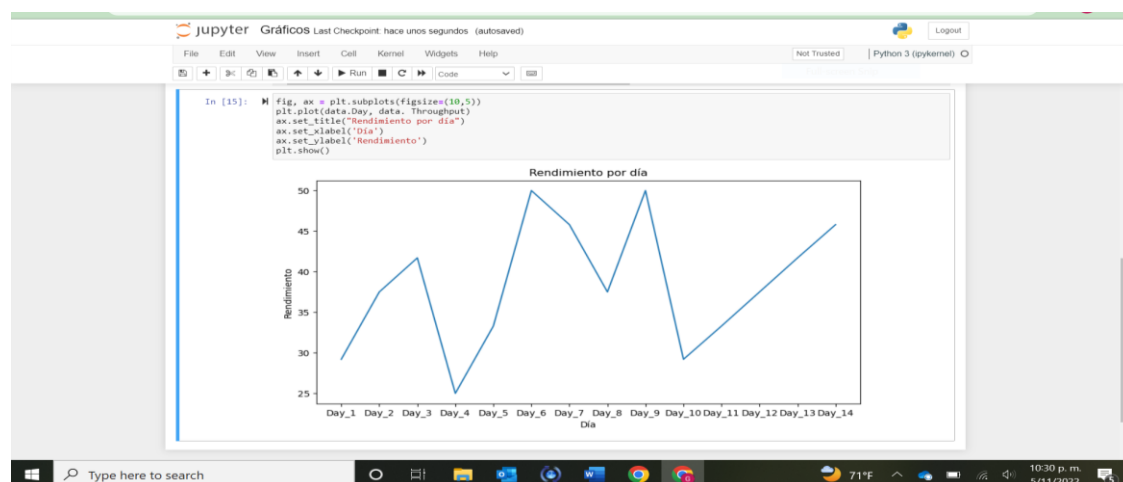
Gráfica N° 2, se aprecia una gráfica de líneas en donde podemos identificar cómo avanza a lo largo de los 14 días el número de unidades producidas. También, deducimos que no hay un balance, hay caídas y subidas muy notables. Los días con mayor pico de producción es el 6 y 9, con 1200 unidades producidas cada una.

3. Gráfica de tiempo



Gráfica N° 3, podemos ver que siempre se mantiene en 24 horas por día para evaluar el número de unidades producidas.

4. Rendimiento



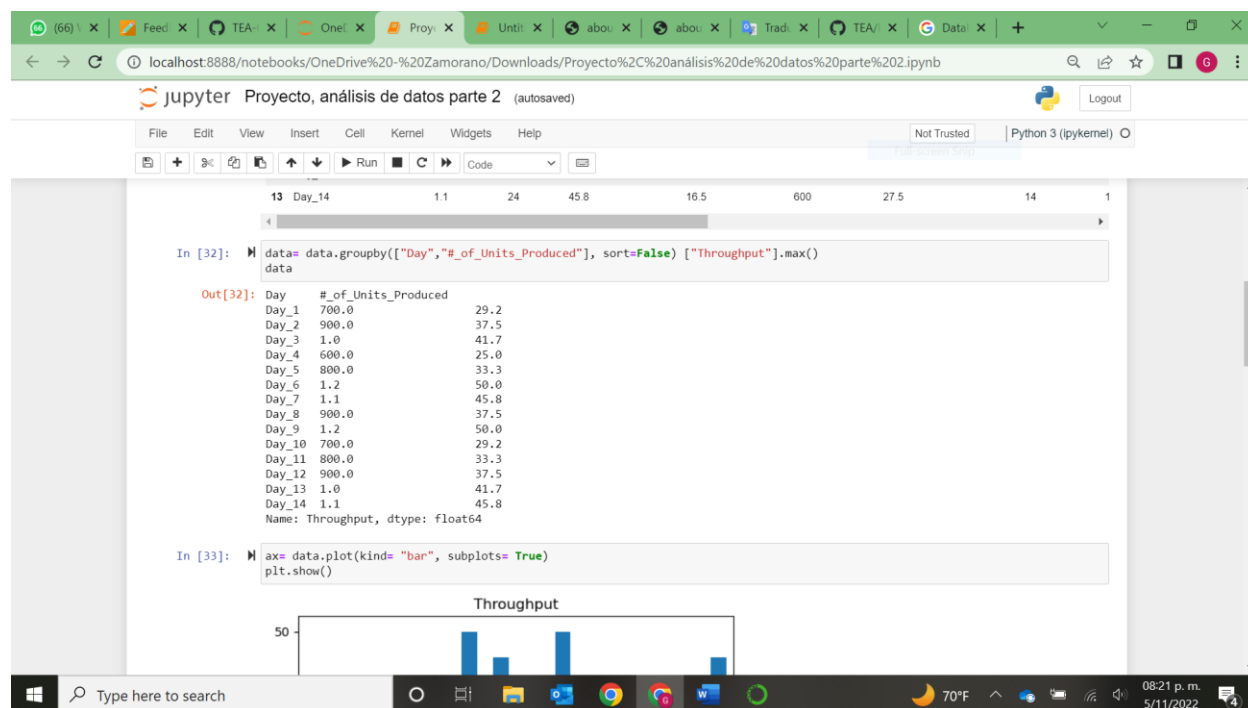
Gráfica N° 4, se puede apreciar una gráfica de líneas, aquí se aprecia en cómo el rendimiento actúa a lo largo de los 14 días, se deduce que hay días con alto grado de conveniencia para el productor, sin embargo, en el día 4 hubo una gran caída en lo que es rendimiento y por supuesto en el número de unidades producidas.

5. Dataframe

El formato de estas estructuras se puede compararse con los diccionarios de Python. Efectivamente, las claves son los nombres de las columnas y los valores son las Series. Su estructura puede considerarse similar a la de una hoja de cálculo de Excel.

- Dataframe de la base de datos del proyecto

Se puede observar que tenemos los datos como días, número de unidades producidas y el rendimiento que se obtuvo, a partir de estas podremos analizar los tres parámetros mencionados al principio de esta presentación y se crearán más gráficos.



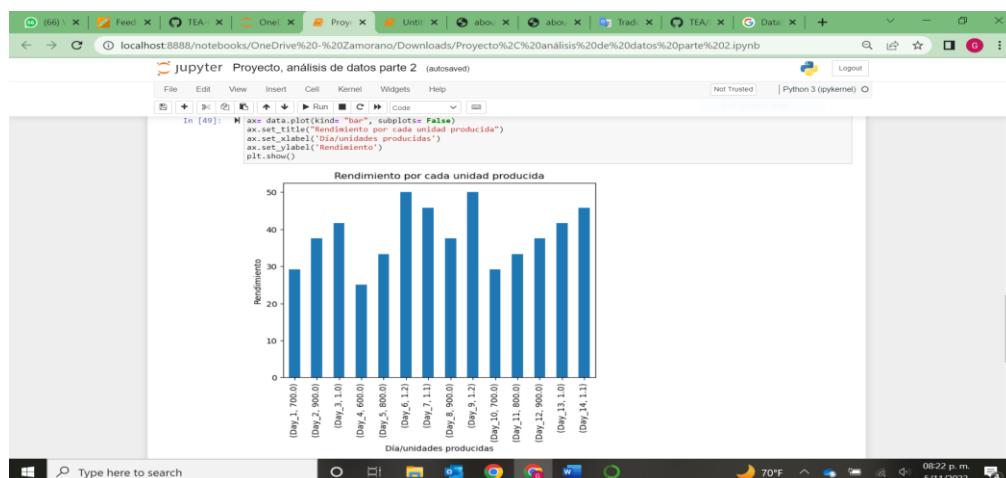


Gráfico N° 5, Esto lo hicimos mediante el parámetro `kind='bar'`. Estamos usando esto porque queremos dar a conocer los días y número de unidades producidas, asimismo, todo esto va en relación con los rendimientos obtenidos en ese día. Básicamente en esta gráfica de barras podemos observar los días cómo actúa el rendimiento al largo de los 14 días.

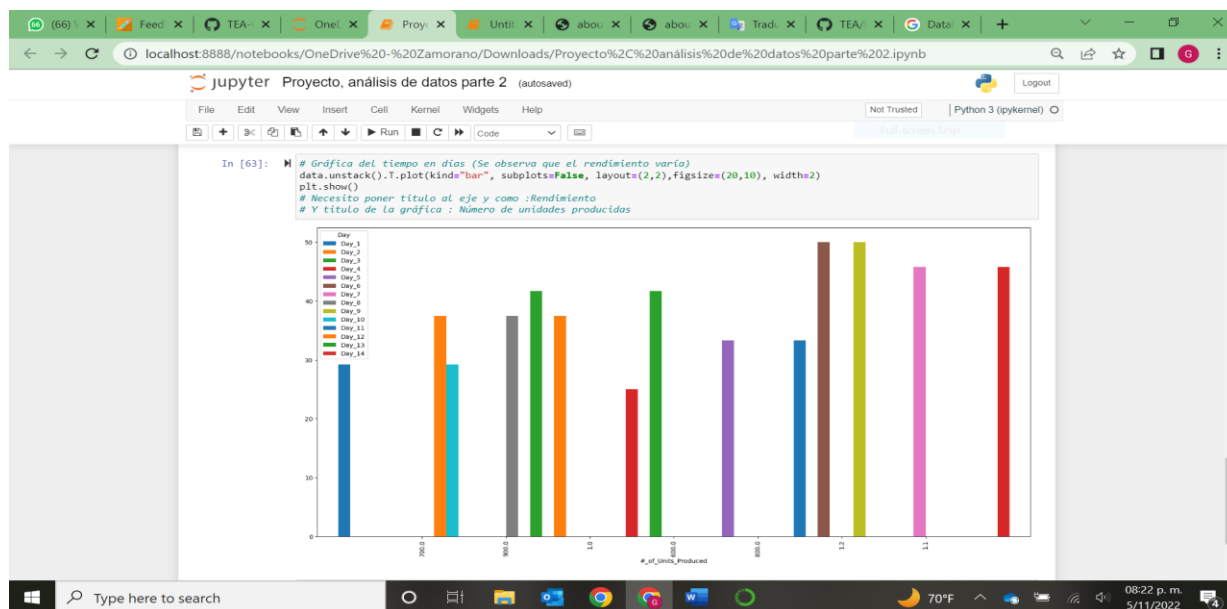


Gráfico N° 6: Se puede observar el número de días representado con varios colores, además en cada uno de ellos, en el eje “X”, se aprecia el número de unidades producidas y en eje “Y” el rendimiento total por día. Asimismo, se observa que el día en que menos

rendimiento fue el día 4 en donde hubo un rendimiento de 25 con 600 unidades producidas. Por otro lado, el día en que mayor rendimiento hubo fue en el 6 y 9, ambas con 1200 unidades producidas y rendimiento de 50.

CONCLUSIONES

- ❖ Ahora podemos analizar cualquier base de datos utilizando los códigos implementados en este proyecto, relacionado a los parámetros a usar en un invernadero. La demostración se encuentra en GitHub, y en las capturas adicionales en este documento, allí se pudo ejecutar los códigos mediante VSC, Python y VSC. Asimismo, en el repositorio se puede observar todo lo que realizamos para la ejecución del proyecto.
- Pudimos analizar los 3 parámetros que establecimos desde un inicio:
 - Número de días establecidos para cosecha de unidades.
 - Número de unidades producidas por cada ciclo del cultivo
 - Rendimiento Total de las producciones

Gracias a los análisis realizados, nos podemos dar cuenta que como productores debemos enfocarnos en tomar datos si es posible diariamente para tener un mejor control de nuestra producción y ventas, de esta manera podemos optimizar nuestros recursos.

- ❖ Logramos manejar las herramientas de los sistemas de información gracias a la electiva brindada en este periodo académico.
- ❖ VSC, Pandas, GitHub, matplotlib, Jupyter Notebook y Phyton son herramientas muy usadas hoy en día y de suma importancia para todas las áreas de producción, asimismo para otras industrias. Debemos aprovechar la tecnología de estas para poder mejorar, tomar control y analizar nuestro sistema de negocios.

REFERENCIAS:

- *Anaconda | Anaconda Distribution.* (2022). Anaconda; Anacoda. <https://www.anaconda.com/products/distribution>
- Alfredo Sánchez Alberca. (2020). *La librería Matplotlib | Aprende con Alf.* Aprende Con Alf. <https://aprendeconalf.es/docencia/python/manual/matplotlib/>
- *BIOENERGÍA Y SEGURIDAD ALIMENTARIA ÉVALUACIÓN RÁPIDA (BEFS RA) Manual de Usuario PRODUCCIÓN DE CULTIVOS.* (n.d.). <https://www.fao.org/3/bp851s/bp851s.pdf>
- *Matplotlib — Visualization with Python.* (2022). Matplotlib.org. <https://matplotlib.org/>
- Miguel Servín Palestina, Leonardo Tijerina Chávez, Guillermo Medina García, Oscar Palacios Velez, & Héctor Flores Magdaleno. (2017). Sistema para programar y calendarizar el riego de los cultivos en tiempo real. *Revista Mexicana de Ciencias Agrícolas*, 8(2), 423–430. <https://www.redalyc.org/journal/2631/263150548014/html/>
- *Pandas - Python Data Analysis Library.* (2022). Pydata.org. <https://pandas.pydata.org/>
- *Project Jupyter.* (2022). Jupyter.org. <https://jupyter.org/>
- Universidad EAFIT. (2022). *Software para invernadero - Innovación EAFIT / Transferencia de Tecnología y Conocimiento - Universidad EAFIT.* Eafit.edu.co; [www.eafit.edu.co](https://www.eafit.edu.co/innovacion/transferencia/Paginas/software-invernadero.aspx); <https://www.eafit.edu.co/innovacion/transferencia/Paginas/software-invernadero.aspx>