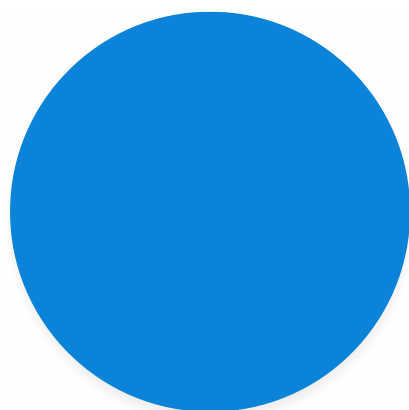
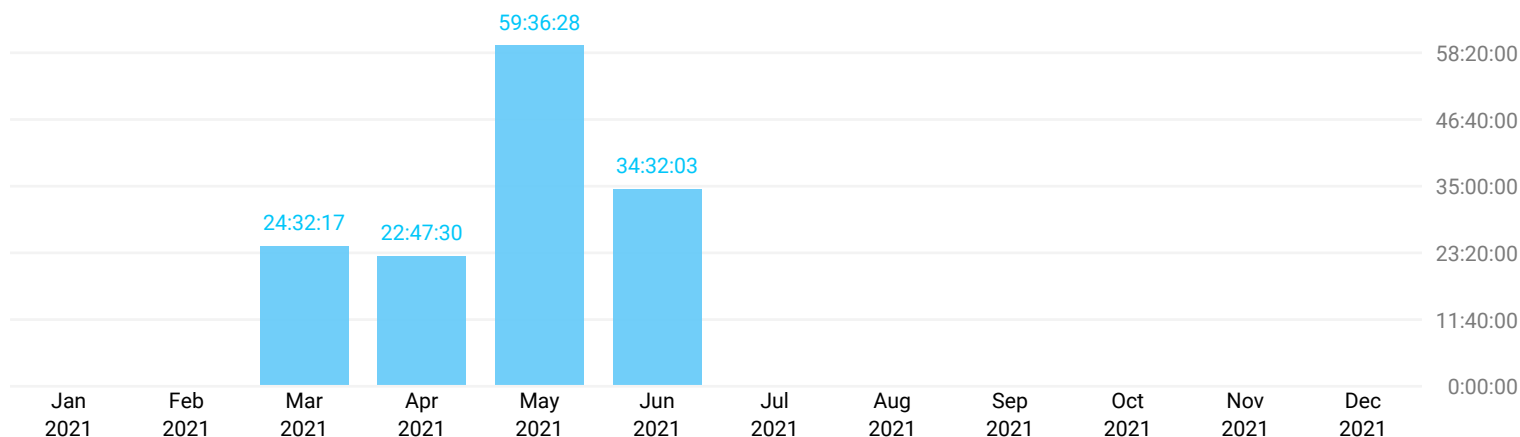


# Summary Report

01/01/2021 – 12/31/2021

TOTAL HOURS: 141:28:18

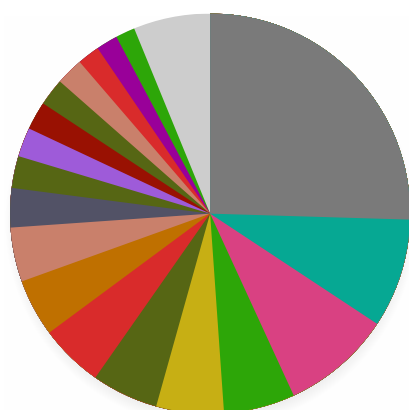


## PROJECT

● SOS2021-04

## DURATION

141:28:18



## TIME ENTRY

- Without description
- #31 Realización D03. Implementación de integraciones, tests de front-end y video explicativo
- #23 Implementar front-end con Svelte
- #21 Implementacion Base de datos en API para trabajar con persistencia
- Debe tener desplegado en Heroku una aplicación funcional que tenga las siguientes características:
- #27 Una forma de buscar recursos utilizando la búsqueda que proporciona la API. (p.e. statistics?from=2000&to=2017)
- #26 Implementar paginacion y detalles de front end no resueltos en la entrega anterior
- La API debe implementar adecuadamente todos los casos de la "tabla azul" comentada en el L06
- #29 Implementar graficas para education expenditures
- #30 Un frontend funcional que tenga, además de las funcionalidades solicitadas en entregables anteriores, una nueva vista con visualizaciones de todos los datos que están en la API con la información a partir de los datos que proporciona la API en un widget de tipo "line" (similar al que se vio en el L11).
- La(s) issue(s) debe estar en estado cerrado antes de la sesión de feedback.

## DURATION

36:05:00  
12:28:03  
12:23:24  
8:15:30  
7:44:00  
7:37:00  
7:13:17  
6:36:59  
6:14:47  
4:23:00  
3:36:53

● Tener al menos 3 horas dedicadas registradas desde la última sesión de feedback.	3:18:15
● #10 Crear al menos una issue en el repositorio del principal del proyecto (SOS2021-XX) reflejando alguna de las tareas de este feedback	3:16:00
● Tener dentro de la carpeta "public" un archivo "index.html" con la misma información que se introdujo en el archivo "Readme.md" del repositorio común.	3:12:21
● Aplicar API en Heroku tal y como se especifica en L06	3:04:53
● #12 Tener una ruta "/info/poverty_risk generada en el servidor NodeJS (NO se permitirá un archivo .html)	2:32:20
● #22 Se debe tener una colección de llamadas de postman con al menos 2 tests sobre la API desarrollada	2:23:46
● #19 Tarea Extra: Aplicar ejercicios Lab 07 al proyecto	2:13:00
● Other time entries	8:49:50

## PROJECT - TIME ENTRY

## DURATION

● SOS2021-04	141:28:18
#10 Crear al menos una issue en el repositorio del principal del proyecto (SOS2021-XX) reflejando alguna de las tareas de este feedback	3:16:00
#12 Tener una ruta "/info/poverty_risk generada en el servidor NodeJS (NO se permitirá un archivo .html)	2:32:20
#15 Haber leído todas las noticias en Piazza previas a las 18:00 horas del día anterior a la sesión de feedback.	1:01:36
#19 Tarea Extra: Aplicar ejercicios Lab 07 al proyecto	2:13:00
#21 Implementacion Base de datos en API para trabajar con persistencia	8:15:30
#22 Se debe tener una colección de llamadas de postman con al menos 2 tests sobre la API desarrollada	2:23:46
#23 Implementar front-end con Svelte	12:23:24
#26 Implementar paginacion y detalles de front end no resueltos en la entrega anterior	7:13:17
#27 Una forma de buscar recursos utilizando la búsqueda que proporciona la API. (p.e. statistics?from=2000&to=2017)	7:37:00
#29 Implementar graficas para education expenditures	6:14:47

# PROJECT - TIME ENTRY

## DURATION

#30 Un frontend funcional que tenga, además de las funcionalidades solicitadas en entregables anteriores, una nueva vista con visualizaciones de todos los datos que están en la API con la información a partir de los datos que proporciona la API en un widget de tipo "line" (similar al que se vio en el L11).	4:23:00
#31 Realización D03. Implementación de integraciones, tests de front-end y video explicativo	12:28:03
Aplicar API en Heroku tal y como se especifica en L06	3:04:53
codigo nodeJS y Express	1:33:21
Configurar README y html en /public	0:57:07
Crear ruta /info/education_expenditure y conjunto de datos en POSTMAN	1:54:31
Debe tener desplegado en Heroku una aplicación funcional que tenga las siguientes características:	7:44:00
Inclusión de 5 ejemplos reales en ficha de datos del proyecto	0:09:45
Instalar y Configurar extension Toggl Button	0:01:34
La API debe implementar adecuadamente todos los casos de la "tabla azul" comentada en el L06	6:36:59
La(s) issue(s) debe estar en estado cerrado antes de la sesión de feedback.	3:36:53
Tener al menos 3 horas dedicadas registradas desde la última sesión de feedback.	3:18:15
Tener dentro de la carpeta "public" un archivo "index.html" con la misma información que se introdujo en el archivo "Readme.md" del repositorio común.	3:12:21
Tener instalado Chrome	0:00:38
Tener una cuenta de GitHub.	0:00:23
Visualización de video e Implementación servidor web con Node.js en	0:58:19
Visualización de video e Implementación servidor web con Node.js en	2:12:36

PROJECT - TIME ENTRY	DURATION
Without description	36:05:00