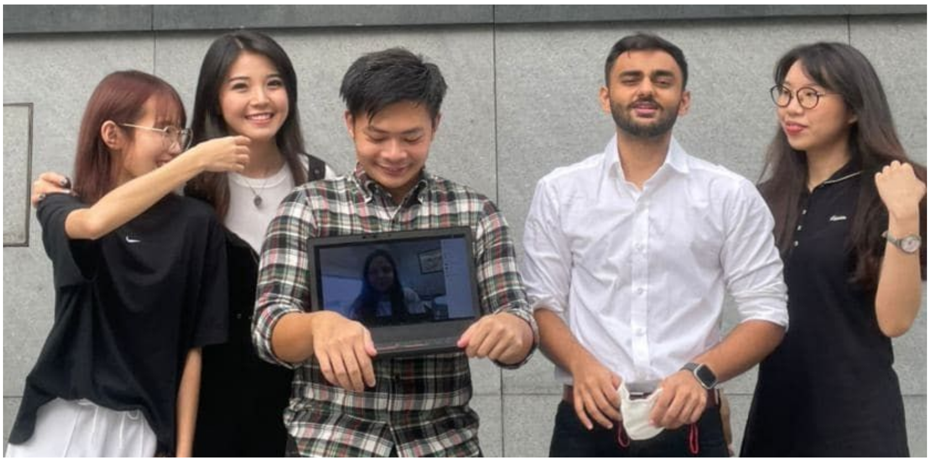
IS483: IS Project Experience (Business Analytics)
AY 2020/2021 Semester 1

Supervisor	Wang Hai
Reviewer & Track Coordinator	TAN Poh Choo & WANG Zhaoxia
Client	Credit Suisse Group Singapore
Project Name	Natural Disaster Stress Testing
Group Name	Cycovid-20



Group Members:

Lim Jia Yu	jiayu.lim.2017@sis.smu.edu.sg
Lyu Xiaowei	xiaowei.lyu.2017@sis.smu.edu.sg
Parth Goda	parthrg.2017@sis.smu.edu.sg
Shivika Khemka	shivikak.2017@sis.smu.edu.sg
Tan Kiang Boon Cedrick	cedrick.tan.2017@sis.smu.edu.sg
Yan Bai Shuang Christine	ybyan.2017@sis.smu.edu.sg

Table of Content

1. Business Overview

1.1 Client Background

Credit Suisse was founded in 1856, started off as a Swiss investment bank, which turned into a success story spanning over the next century and a half, with Credit Suisse gradually evolving into a leading global provider of financial services.

This has involved the amalgamation of very different cultures, philosophies, and spheres of specialist knowledge, which has resulted in the creation of a strongly integrated bank. As of 2006, the bank began operating as a globally active integrated universal bank providing comprehensive solutions to our clients in private banking, investment banking, and asset management.

1.2 Project Task and Value Proposition

The project aims to develop an interactive and user-friendly web dashboard that provides users with an understanding of the potential financial risks that assets undergo during a cyclone/typhoon within a selected country in the APAC region.

Using the application, our client will be able to understand the financial risks of past cyclones/typhoons, identify “at-risk” geographical locations as well as simulate their own scenarios to provide them an edge in prudent business decision making.

Further, the application has been simplified (as per client’s request) such that a user without industry background will still be capable of using the application

1.3 Evidence of Deliverables

The team came up with a Poster to summarise our project work. All the requirements which include Poster, Presentation Slides, pitch video as well as the Project Report are located in the same google docs folder for submission. Here is the link to the google docs folder:
https://drive.google.com/drive/folders/1e6_IXjsInDkL5zszAK_diUxjLcMEZ65U?usp=sharing

2. Research Study and Preliminary Analysis

The project surrounds the problem statement: What is the impact of typhoons on the economy under the effects of climate change? Therefore, the team has conducted a research study to better understand the problem and provide suitable solutions.

2.1 Country Selection

The team’s focus was to work on countries in the APAC region. The team looked for just two countries to focus on on the basis of:

- High frequency of typhoons making landfall
- A significant number of these typhoons be of intensity between category 1 to 5
- Data of such typhoons is recorded and accessible

At the same time, the team wanted the countries to be of two different development levels:

- Developed Countries
- Less Developed Countries

After finding data from IBTrACS for every typhoon from 1852-2018, the team visualized the number of typhoons in each country and got this map. Initially, the selections were Japan and the Philippines based on the frequency heat map generated from PowerBI *(Figure 1).*

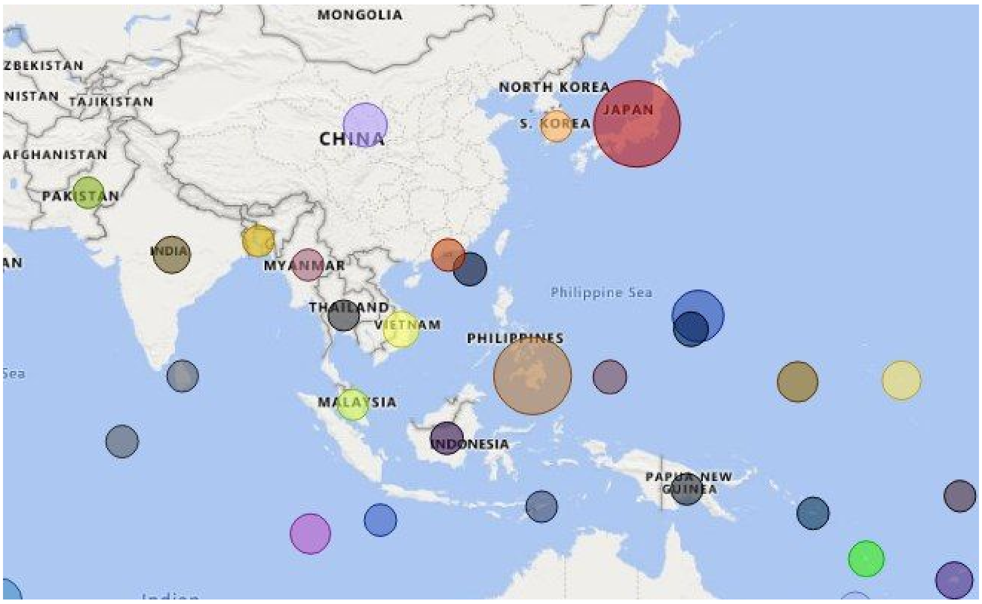


Figure 1: Frequency of typhoons in APAC region

2.1.1 Vulnerability Index: Macro-economic

However, the team had to look deeper. First, the team listed out all the countries in the APAC region and ranked them by their GDP per capita growth rate and their current population density. The next step was to identify which of these countries were more prone to damage and slow to recover. It would have been very costly and would have taken a lot of time to do individual research on each country in the APAC to find their risk and development levels. Therefore, the team researched as many world governance indices that were available and relevant. Out of all, the Global Sustainable Competitiveness Index and World Risk Index were the clear choices.

Global Sustainable Competitiveness Index (GSCI) measures and ranks countries based on their competitiveness and sustainability. Looking beyond GDP and other economic or financial measures, GSCI amalgamates the country's environment, society, government, resources, and economy to create a well-rounded index.

On the other hand, the World Risk Index (WRI) measures a country's ability to handle turmoil. It calculates a number based on a country's exposure to natural disasters, how well the country is built, a government's coping capability, and its investment into the future.

With these two indices, the team ranked the APAC countries and came up with the following weights for each of the macro-economic factors for a final number.

Factor	Weight	Reason
Average GDP per capita	0.25	Countries with high GDP per capita are less vulnerable to lasting damages from typhoons. They are able to recover faster financially. These countries also have stable and well-structured economies. The opposite is also true for places with low GDP per capita.
Population Density	0.5	The denser the country, the higher the mortality rate per storm. Human capital is the most impactful for an economy as it not only a financial aspect but a psychological one too. If the death toll is too high, the country loses its bearings and needs a lot of time and resources to recover.
GSCI	0.125	Both Indices are given an equal weightage. The GSCI measures the country's current abilities.
WRI	0.125	The WRI measures its preparations for the future or the worst possible scenarios.

2.1.2 Vulnerability Index: Typhoons

The second part of this index was how long, on average, a country was hit by typhoons and the average category of all those typhoons. Taking the data from IBTrACS, the team filtered the data all the way down to just these two articles and ranked each country accordingly. A higher rank for countries that do not get hit so often as well as those with a low average category. The team gave each of these two rankings a 0.5 weightage to come up with a final number.

2.1.3 Vulnerability Index: The Final Formula

Vulnerability of each country = 0.5 (0.25 * Average GDP per capita +0.5 * Population Density + 0.125 * GSCI + 0.125 * WRI) + 0.5 (0.5 * Duration + 0.5 * Category)

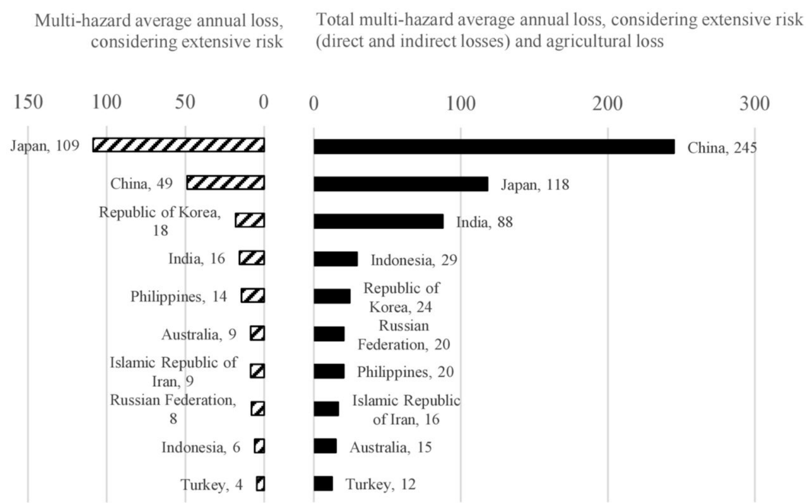
All the countries were then ranked according to this number (after normalization), and the following shows here were the bottom 10 (most vulnerable) countries *(Figure 2)*.

23	India	2.548	31	16.22495145	4	38.4	27	29.36	29	1.80	10	2.55	9	16.75	9.5	26.25	0.5858208955	22	India
24	Indonesia	2.963	14	73.04003185	13	39.5	24	6.77	14	1.52	15	3.00	12	14.75	13.5	28.25	0.6455223881	23	Indonesia
25	Kiribati	-0.522	27	24.00984413	6	34.4	31	56.71	31	1.49	11	3.00	11	17.5	11	28.5	0.6529850746	24	Kiribati
26	Vanuatu	0.399	12	454.9380726	28	45.4	10	10.58	20	1.58	8	2.47	8	20.75	8	28.75	0.6604477612	25	Vanuatu
27	Bangladesh	1.127	26	1239.579312	31	39.4	25	18.78	25	0	1	0	1	28.25	1	29.25	0.6753731343	26	Bangladesh
28	Japan	3.819	24	24.99160625	7	41.9	18	20.69	26	1.76	16	3.87	13	15	14.5	29.5	0.6828358209	27	Japan
29	Samoa	1.901	25	56.18607748	12	40.2	22	22.18	28	2.39	12	6.00	17	18.5	14.5	33	0.7873134328	28	Samoa
30	Papua New Guinea	1.133	9	270.9930748	25	51.1	2	9.19	18	1.59	14	6.67	18	17.25	16	33.25	0.7947761194	29	Papua New Guinea
31	Solomon Islands	-0.778	30	90.29941666	16	35.5	30	14.64	21	1.57	7	5.50	16	21.875	11.5	33.375	0.7985074627	30	Solomon Islands
32	Philippines	1.221	18	498.6598703	29	36.2	29	6.19	12	1.87	17	4.16	15	24.125	16	40.125	1	31	Philippines

Figure 2: Top 10 vulnerable countries

From there, the team chose to focus on Japan as the developed country and the Philippines as the developing country. Solomon Islands, Samoa, and Papua New Guinea were not selected due to their small populations and very low presence of financial institutes.

However, based on research, the team noticed that India ranked much higher on yearly financial loss due to natural disasters such as typhoons *(Figure 3)*. By selecting India, the team can also focus on a different water body to include the Indian Ocean other than the South Pacific Ocean. More importantly, the client is interested in northern India due to the recent Cyclone Amphan. Final selections of Japan and India were made from the table below *(Figure 4)*.



Source: *Asia-Pacific Disaster Report 2019*.

Figure 3: Annual loss due to cyclones

	AVG GDP per capita rate	Rank	Pop density	Rank	Global $	GSCI	GRI	GRI F	AVG (	CAT	AVG DI	DUR/	Governanc	Cyclone	Final score	Normailized	Final rank
India	2.548	31	16.22495145	4	38.4	27	29.36	29	1.80	10	2.55	9	16.75	9.5	26.25	0.5858208955	22
Indonesia	2.963	14	73.04003185	13	39.5	24	6.77	14	1.52	15	3.00	12	14.75	13.5	28.25	0.6455223881	23
Kiribati	-0.522	27	24.00984413	6	34.4	31	56.71	31	1.49	11	3.00	11	17.5	11	28.5	0.6529850746	24
Vanuatu	0.399	12	454.9380726	28	45.4	10	10.58	20	1.58	8	2.47	8	20.75	8	28.75	0.6604477612	25
Bangladesh	1.127	26	1239.579312	31	39.4	25	18.78	25	0	1	0	1	28.25	1	29.25	0.6753731343	26
Japan	3.819	24	24.99160625	7	41.9	18	20.69	26	1.76	16	3.87	13	15	14.5	29.5	0.6828358209	27
Samoa	1.901	25	56.18607748	12	40.2	22	22.18	28	2.39	12	6.00	17	18.5	14.5	33	0.7873134328	28
Papua New Guinea	1.133	9	270.9930748	25	51.1	2	9.19	18	1.59	14	6.67	18	17.25	16	33.25	0.7947761194	29
Solomon Islands	-0.778	30	90.29941666	16	35.5	30	14.64	21	1.57	7	5.50	16	21.875	11.5	33.375	0.7985074627	30
Philippines	1.221	18	498.6598703	29	36.2	29	6.19	12	1.87	17	4.16	15	24.125	16	40.125	1	31

Figure 4: Vulnerability ranking

2.2 Data Collection & Analysis

The data collection process for the research portion was very exhaustive. The team first started with the IBTrACs dataset as a foundation for the research, and then added on to it by using the various news outlets and organizations to collect specific information about each typhoon that occurred in Japan or India between 2013-2020.

One of the main data challenges the team had faced was how dispersed the information was. There was no one repository for typhoon damages and little information on past typhoon damage values and recovery periods. Most news sources focused on the death toll and the areas affected. To help with modelling and frameworks, Credit Suisse provided a previous study based on earthquakes, their damages, and recovery durations. However, typhoons' data was unlike the earthquakes'. Thus the team was unable to find much data on economic recovery and damages.

Here is a list of some of our sources and what obtained from each source:

Source	**Information Acquired**
AON damage reports	Financial and monetary information regarding damages done by typhoons
Various News Articles	Find information about death toll, damages and infrastructure
Research Papers	Primary source of information on climate change, typhoons and other scientific concepts
Japanese Meteoric Society	Typhoon related information like Category, wind speed and duration of each typhoon in Japan
IBTRACS	The Database used to plot typhoons around the world in the application. Also used to get information about their frequency, duration, and categories.
Insurance reports	Supplement information from damage reports
WHO	Damage reports for massive typhoons
Climada Library	Damage models for natural disasters
TradingView	For analysis of effects of the typhoons on stock indexes
Worldbank	Damage reports for massive typhoons
Trading Economy	For data on annual GDP in Japan and India
OECD	Damage reports for massive typhoons
Credit Suisse Earthquake Report	How previous models were designed and how to model our typhoon damage functions

All this data was cleaned and was then placed in an excel data table *(Figure 5)* so that the team could use it to derive our damage function, which will be explained in the latter part.

Cyclone Name	Year	Category (SSHWS)	Damge ($ USD)	GDP (USD$)	Percentage Damage	GDP annual growth (%)	GDP per capita (USD)
Nisarga	2020	1	817,500,000.00	2,610,000,000,000.00	0.03132%	-23.900	1,900.00
Gaja	2018	1	775,000,000.00	2,713,170,000,000.00	0.02856%	5.024	2,169.10
Mora	2017	1	297,000,000.00	2,652,750,000,000.00	0.01120%	5.024	2,169.10
Helen	2013	1	796,000,000.00	1,856,720,000,000.00	0.04287%	6.386	1,544.60
Lehar	2013	1	671,375,000.00	1,856,720,000,000.00	0.03616%	6.386	1,544.60
Hikka	2019	2	1,000,000.00	2,875,140,000,000.00	0.00003%	5.024	2,169.10
Vardah	2016	2	3,370,000,000.00	2,294,800,000,000.00	0.14685%	8.256	1,875.70
Madi	2013	2	1,685,500,000.00	1,856,720,000,000.00	0.09078%	8.256	1,875.70
Bulbul	2019	3	337,000,000.00	2,875,140,000,000.00	0.01172%	8.256	1,875.70
Maha	2019	3	631,035,000.00	2,875,140,000,000.00	0.02195%	6.386	1,544.60
Vayu	2019	3	140,000.00	2,875,140,000,000.00	0.00000%	7.996	1,751.70
Titil	2018	3	1,267,000,000.00	2,713,170,000,000.00	0.04670%	5.024	2,169.10
Ockhi	2017	3	920,000,000.00	2,652,750,000,000.00	0.03468%	5.024	2,169.10
Fani	2019	4	8,100,000,000.00	2,875,140,000,000.00	0.28173%	5.024	2,169.10
Hudhud	2014	4	3,580,000,000.00	2,039,130,000,000.00	0.17557%	8.256	1,875.70
Nilofar	2014	4	5,840,000,000.00	2,039,130,000,000.00	0.28640%	6.386	1,544.60
Amphan	2020	5	13,900,000,000.00	2,610,000,000,000.00	0.53257%	-23.900	1,900.00
Kyarr	2019	5	8,000,000,000.00	2,875,140,000,000.00	0.27825%	5.024	2,169.10
Phailin	2013	5	4,260,000,000.00	1,856,720,000,000.00	0.22944%	8.256	1,875.70

Figure 5: Cleaned data table

2.3 Case Study: India

The team had decided to do some in-depth analysis of India and the Bay of Bengal. This decision was made after finding out information from multiple sources that the Bay is going to experience more frequent *(Figure 6)*, more intense storms *(Figure 7)*, and longer cyclones *(Figure 8).*

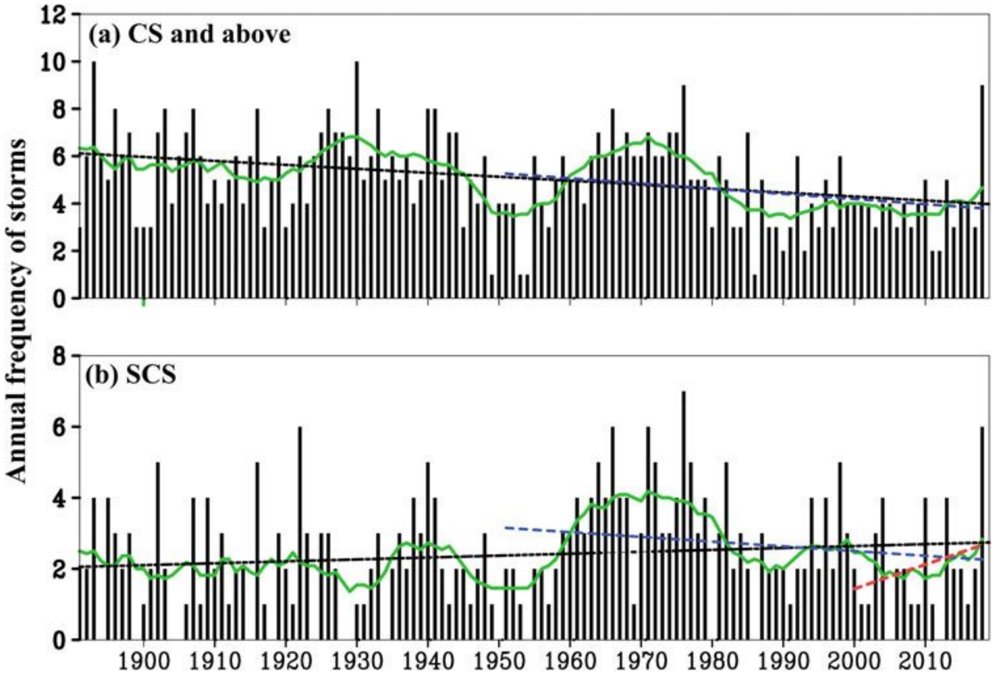


Figure 6: Observed annual frequency of a) Cyclonic Storm and above b) Severe Cyclonic Storm in the NIO. Overall linear trend is indicated by black dashed line, 2000-2018 represented by red dashed line, 1951-2018 represented by blue dashed line

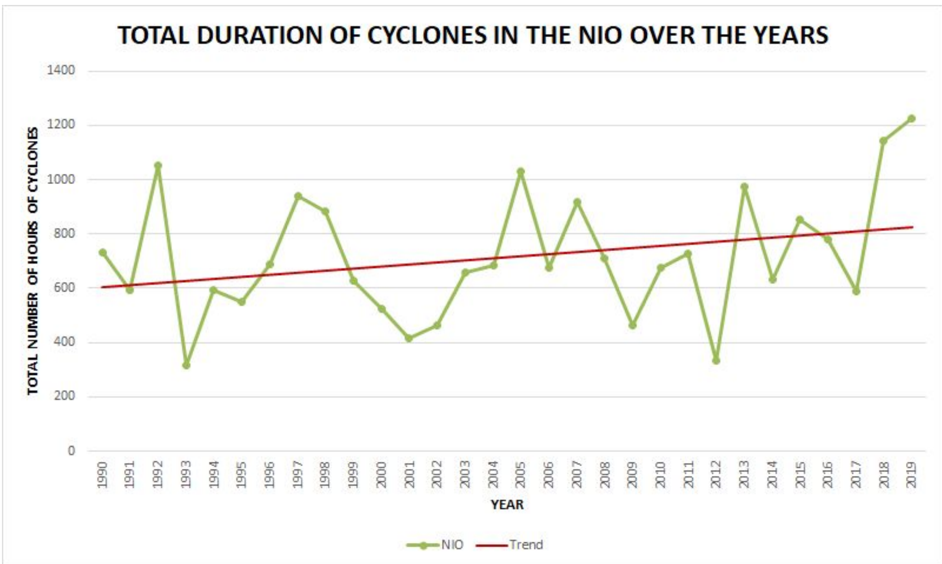


Figure 7: Duration of cyclones in Northern Indian Ocean

Year	Category—basin—month—name of the storm
2000	VSCS—BOB—Nov—BB 05; VSCS—BOB—Dec—BB 06
2001	ESCS—AS—May—ARB 01
2003	VSCS—BOB—May—BB 01
2004	VSCS—BOB—May—BB 02
2006	VSCS—BOB—Apr—Mala
2007	VSCS—BOB—Nov—Sidr
2008	ESCS—BOB—Apr—Nargis
2010	ESCS—BOB—Oct—Giri
2011	VSCS—BOB—Dec—Thane
2013	ESCS—BOB—Oct—Phailin; VSCS—BOB—Nov—Lehar; VSCS—BOB—Dec—Madi
2014	ESCS—BOB—Oct—Hudhud; ESCS—AS—Oct—Nilofar
2015	ESCS—AS—Oct—Chapala; ESCS—AS—Nov—Megh
2016	VSCS—BOB—Dec—Vardah
2017	VSCS—BOB—Nov—Ockhi
2018	ESCS—AS—May—Mekunu; VSCS—AS—Oct—Luban; VSCS—BOB—Oct—Titli

Source IMD

Figure 8: Extreme category cyclones observed over the Indian seas from 2000-2018

This is caused and amplified by the rising sea levels and rising ocean temperatures around the Bay of Bengal. Due to climate change, the temperature in the western Indian ocean is clearly increasing over the last few years, and this will lead to more intense cyclones forming at a faster rate, which was demonstrated by cases of Phailin and Amphan. Also, the warmer oceans will also lead to higher rainfall during storms which in turn will contribute to the increasing intensity of tropical cyclones. The rising sea levels will also pose an immense threat to the coastal areas and exploding population in these areas which will lead to greater damages recorded. India's expanding fragile infrastructure will vastly increase the vulnerability of the areas too.

To illustrate the point and to conduct a research analysis of cyclones in India, the team has chosen two cyclones about seven years apart. The first cyclone the team aims to analyze would be the recent Cyclone Amphan that hit India and countries around the Bay of Bengal on 20 May 2020. The second cyclone would be Cyclone Phailin that hit India back on 12 October 2013. Both the cyclones are Category 4 cyclones with a similar wind speed of 250-260 km/h. In this section of the report, the team aims to analyze and compare the extent of the damages as well as the initiatives that the government took to mitigate the situation.

2.3.1 Cyclone Amphan

Cyclone Amphan was a powerful and catastrophic tropical cyclone that caused widespread damage in Eastern India, specifically West Bengal, and also Bangladesh in May 2020 with the highest wind speed reaching 260 km/h (highest).

s/n	Sector	Extent of assessed damage so far	Quantum of Damages (USD-Billion)
1	Dwelling Houses	28.56 lakhs houses damaged	3.87

2	Agriculture	17 lakh hectares agriculture land crops - Boro paddy / Moong / Til / Jute / Groundnut / Sugarcane / Maize / Cotton	2.15
3	Horticulture	Area - 250556.17 Ha Betel vine, Litchi, Mango etc	0.89
4	Fisheries	Boats - 8007, Huts-1.48 Lakh	0.27
5	ARD	Animals lost - 21.22 lakhs	0.06
6	Drinking-Water	Piped Water Schemes affected 1192	0.28
7	Roads including rural roads and culverts / bridges	Roads - 2148.22km , bridge &CCulverts - 355, Rural Road - 10091.17 Km	0.30
8	Irrigation canals / Ponds	Embankments - 244.73Kms, Sea Dykes - 3.6 KM	0.40
9	Power	Power Substations damaged: 273 poles - 4,49,174	0.44
10	Forest	Forest area affected 1.58 lakh hectares	0.14
11	Education Infrastructure	14,640 schools, 301 colleges	0.11
12	Health Infrastructure	PHC-563, BPHC/RH-169, Sub centre-5142, SDH/SGH-37, DH-24	0.17
13	Anganwadi infrastructure	ICDS Centre damaged - 12678	0.05
14	Urban Infrastructure	Municipal Roads, Street Lighting, Underground Sewerage system, storm, water drainage, water supply schemes, roads, etc	0.92
15	Industries including MSME	Industrial Warehouse, Raw material / Industrial Infrastructure / Sheds	3.64
16	Miscellaneous	Transport, Fire & Emergency Infrastructure, Godowns, Housing, Correctional Homes, BCW etc	0.2
		TOTAL	**13.9 Billion**

2.3.2 Cyclone Phailin

This was the strongest cyclone to hit India since the 1999 Cyclone Odisha. On 11 October, the wind speeds of the cyclone hit 260km/h making the cyclone a Category 5 cyclone. In summary, Cyclone Phailin had a death toll of 44, 256,600 homes damaged and affected about 13.2 million people from 18,370 villages.

s/n	Sector	Extent of assessed damage so far	Quantum of Damages (USD - Million)
1	Dwelling Houses	256,633 homes were affected in which 57,996 were fully damaged.	477.44
2	Public Buildings	12,811 public buildings were damaged. 12,296 were non-residential and 515 residential buildings	106.76
3	Heritage Monuments	Out of 218 monuments protected by the state, only 5 were impacted and assessed	0.008065
4	Roads	1,936.36km of roads in cities and towns 8,099 km in Rural areas	113.03
5	Infrastructure	245.1km of Urban roads 66.6 km of Drains 250 Street lights 44 Water Supply Sources 32.5 Km of Pipes	75.81
6	Agriculture	Out of 4,260,052 hectares of agriculture, horticulture and perennial crops 1,292,967 hectares affected. 782,989 hectares of these hectares saw crop loss more than 50%	287.71
7	Horticulture	Out of 1,205,900 hectares of horticulture 43,358 hectares were affected	25.14
8	Irrigation Network	Extensive damage to: 1. 1,088 km of canals 2. 8,152 minor irrigation projects 3. 4,848 minor lift irrigation projects 4. 1,071 km of river/saline embankments 5. 744 km of drainage channels 6. 641 deep bore wells 7. 399 departmental buildings	110.13
8	Livestock	7.02 million livestock were affected. Out of which 179,518 died in the storms. However, more damage was reported in the buildings and shelters that held the livestock	4.42
9	Fisheries	8,423 Boats and 33,398 Nets lost. Smaller losses in Fish Tanks, ponds, and seeds.	97.53

10	Power	Most damage occurred to power lines (16,442 Km) and 5782 transformers	289.8
11	Business	1564 Artisans and 1039 SMEs affected	2.26
		Total	**$1,469 Million**

2.3.3 Comparisons Between the Two Cyclones and Evaluations

Let's compare the two cyclones based on their important characteristics.

Factor	Amphan	Phailin
Category	5	5
MAX wind (km/h)	260	260
Death Toll (ppl)	128	44
Evacuations (ppl)	2,000,000	550,000
Year	2020	2013
Global Temperature increase (°C)	0.99	0.68
Sea level increase (mm)	95.4	65
Damage (USD $ in billion)	13.9	1.469

In seven years, the global temp rose by 0.3°C, the sea level increased by 30 cm, and the damage was recorded to be tenfold. As explained, there is definitely correlation and causation here between damage and climate change. What's more alarming is that the damage as a percentage of GDP had increased by 10 times too. Stepping away from the monetary terms, India also had a 3 times larger death toll and 4 times the evacuated population. The need for an application to visualize these effects is the main reason for our FYP.

2.4 Climate Change and Cyclones

Let's start with understanding how cyclones are formed. It all starts with a stark difference between the surface temperature of the sea and the upper air. The air closer to the surface is warm, and the air in the upper atmosphere is cool. Similar to a convection current in an enclosed area, the warm moist air rises. More warm air rushed in to fill the space of the rising air. Rising warm air condenses into clouds. Heat is released back into the environment. This process loops into a cycle aided by the earth's rotation. The storm now starts to spin. Refer to the drawing *(Figure 9)* below to visualize this formation.

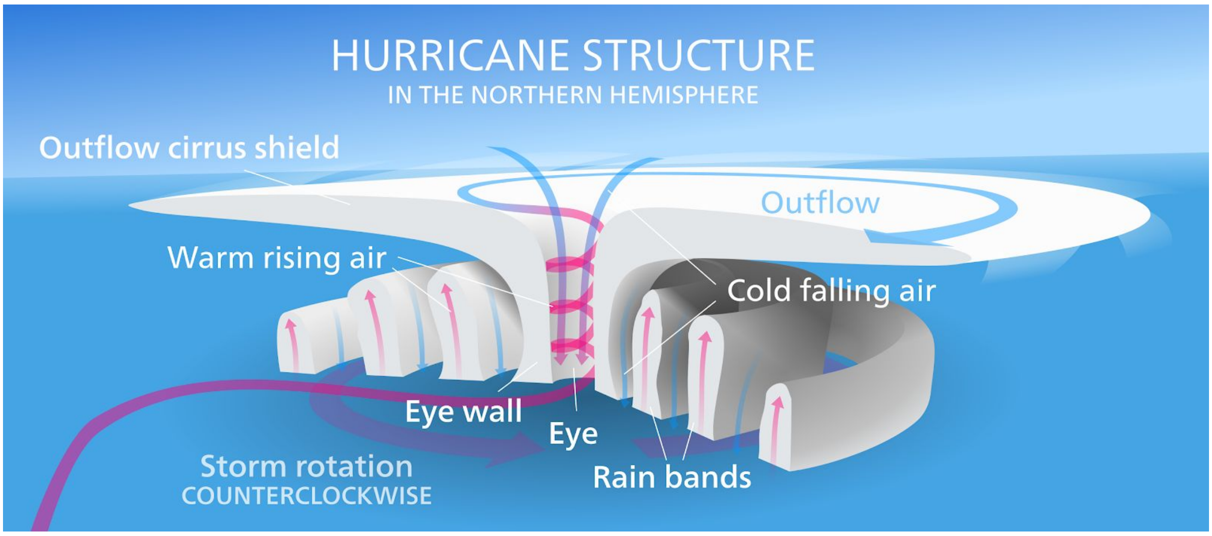


Figure 9: Visual guide to the formations of cyclones

The storm derives its energy from the difference in temperature between the start and endpoints of the cycle. If the sea and its surface temperature rise, there is a greater difference in this temperature. Hence storms grow to become more intense, last longer and bigger.

2.4.1 Factors that Affect Cyclones

Due to the warming of the upper atmosphere and the ocean by greenhouse gases, there is a more stable ocean environment and a lesser chance of cyclones forming from the convection currents in the air. However, when a period of cold winds and weather, the difference between the sea and atmosphere becomes too tremendous and leads to stronger cyclones.

Another human-based activity that significantly affects the characteristics of the formation of cyclones is particulate pollutants and aerosols discharge. These big particles in the atmosphere reflect sunlight away, cooling the environment. A drastic increase or decrease in these particles destabilizes the oceans and creates a temperature difference due to the change in the amount of sunlight absorbed by the sea.

Diminishing man-made aerosols is one of the reasons for the active tropical cyclones in the North Atlantic over the last 40 years. During the end of this century, climate experts predict unprecedented levels of increase in the frequency of cyclones forming. This is due to the cooling effects of the particles.

Surprisingly, volcanic eruptions also play a significant role in the locations, intensity, and frequency of cyclones. Proven by data collected in previous major volcanic eruptions such as Pinatubo in the Philippines in 1991 and El Chichón in Mexico in 1982, cyclones become very volatile and intense when they occur around an eruption. Global warming amplifies these effects.

2.5 Damage Function

To calculate the expected damage that a typhoon can cause, a damage function was generated. The function of Japan is shown below *(Figure 10)*. The x-axis is the category of the typhoons, and the y-axis is the expected percentage of damage.

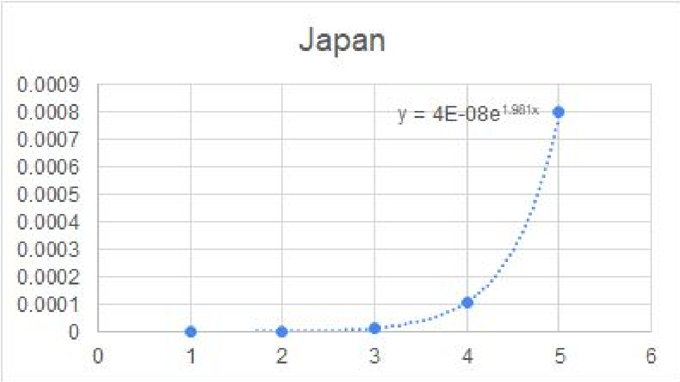


Figure 10: Damage function for Japan

To arrive at these values, the team researched the typhoons that hit Japan from 2013 to 2020 and created a database of the damage caused by these typhoons. The percentage of damage to GDP caused by each of them was calculated and the values then averaged over the category to get one percentage value for each category. The team noticed that these values followed an exponential trend from Category 1 to Category 5, so a best-fit exponential line was generated and the numbers were fit to it.

The "damage" referred to here is the damage to produced capital. Since GDP is a measure of the goods and services produced in the country and produced capital is the machinery, buildings, and equipment in which these goods and services are produced, the team feels that an estimated impact to GDP will be a good estimate for produced capital as well.

To calculate the value damage, the prefectures that were hit by the typhoon and the category of the typhoon when it hit that area are identified. Then, the corresponding percentage from the function is multiplied with the value of produced capital in the prefecture. To calculate the value of produced capital in each prefecture, Climada's exposure to asset data was used. Finally, these values are added for each prefecture to get the total value damage for the typhoon.

{Produced Capital in Pref 1 x Damage %[Category in Pref 1] + Produced Capital in Pref 2 x Damage %[Category in Pref 2] + …} = Total Value Damage

2.6 Evaluations

2.6.1 Climate Change Evaluation

As the temperature of the sea is predicted to rise year on year and the greenhouse effect brings the upper atmosphere temperature higher *(Figure 11)*, the current consensus of the National Oceanic and Atmospheric Administration. This would create a stable atmosphere where the formations of cyclones would not be as favorable. As explained before, while this might cause a decrease to little change in the frequency of the cyclones, their intensities are predicted to increase by up to 10%. Government and financial institutions need to be aware of these models when they make their risk assessment of companies and investments in the northern Indian Ocean and the Bay of Bengal. They need to realize and visualize the correlation between the rising sea level and the damage done in the region. Just a 20 year difference and a 0.60°C difference in temperature (1999 to 2020, 0.39 to 0.99) caused a $9.2billion USD difference in damages. Partly due to the expansion of cities, population growth, and the explosion of businesses and commerce, it's no doubt that climate change has increased a cyclone's ability to devastate everything in its path.

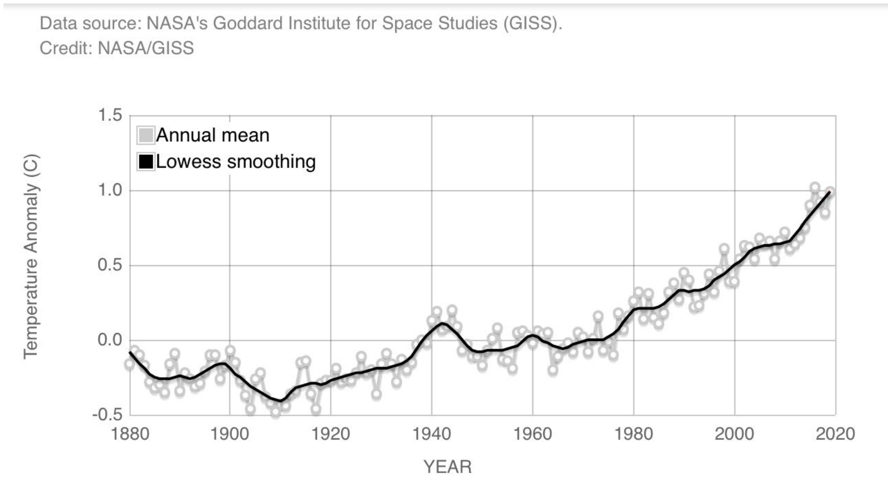


Figure 11: Global temperature anomaly (yearly)

2.6.2 Financial Evaluation

Banks and financial institutions need to use visualizations to simulate and model more intense scenarios to predict the damages that will hit the areas surrounding the Bay of Bengal. They need to prepare for higher liquidity risk and market risks surrounding businesses that originate from these areas. For example, West Bengal is the biggest producer of rice in India, the second-largest potato producer, and owns 30% of the tea production in India. All these industries are heavily dependent on perfect weather and soil conditions. Conditions that are disrupted by cyclones. By being qualitative prepared, financial institutions can leverage their model data to find competitors and alternatives to invest in during the time of crisis. They would know to protect their investments and pull out of commitments in the Bay area. They could take advantage of the shortages in the production of these food types and reinvest when the storm has passed. At this point, land, factories, and warehouses would be dirt cheap due to some damage. With little to minimal operational investment, commodity trades could ride on the area's efforts back to normality and make millions of their investments.

To illustrate the point regarding market risk, the team has taken the candlestick charts of the two biggest stock indexes in India and Japan. On the right side, we have the Nikkei 225 from Japan *(Figure 12)*. Each blue line represents a typhoon making landfall in Japan this year between May till now. The team calculated the average daily change during each typhoon, and it turned out to be about -1.5%. This phenomenon was also observed in NIFTY 50 *(Figure 13)*, the biggest stock index in India.

Figure 12: Nikkei 225. Average change during typhoon season: -1.49% daily

Figure 13: NIFTY 50. Average change during cyclone season: -1.612% daily

However, the focus should be on Credit Suisse and why we need this research and a visual dashboard for them. After all the research and evaluations, the team needs someplace to visualize the damage functions and climate change assessments. The team needed to find a place to quantify these values and churn out recommendations. The risk team at Credit Suisse needs a tool to help them prepare and predict damages and risks in the financial/economic aspects of an incoming typhoon or a cyclone. For that, they need to refer back to historical data and look at previous cyclones in one easy to use dashboard.

3. Solution Approach

3.1 Solution Design

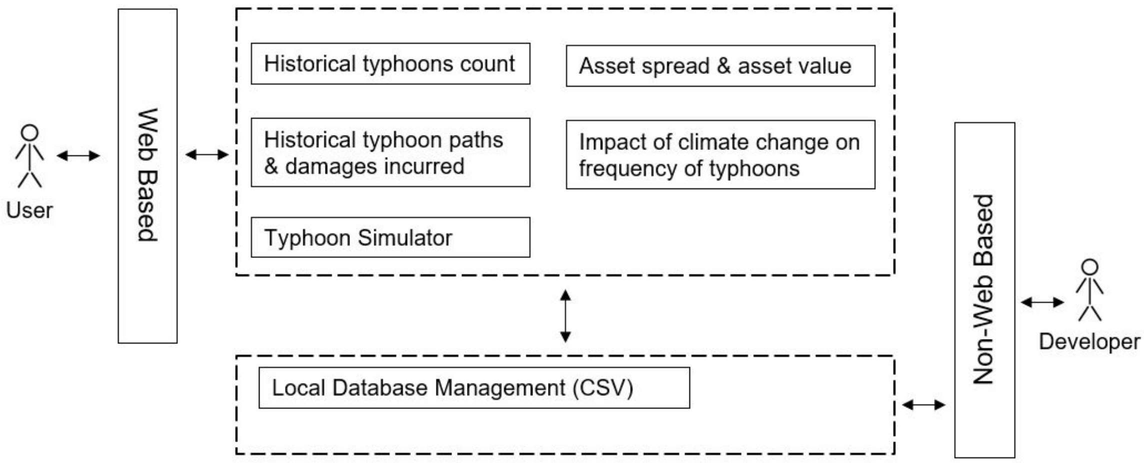


Figure 14: Solution design of the team's application

3.2 Product Features

Our proposed solution enables users to:

- Visualise historical typhoon paths and the damages incurred
- View historical typhoons count based on category by selecting the years
- Easily view asset spread across the country
- Assess the impact of climate change on frequency of typhoons
- Simulate typhoons and assess their damages

3.3 Use Case Diagram

Based on the product features mentioned above, the application contains 5 main components whereby users would be able to view historical typhoon counts for both total and category counts upon selection of the year. Users will be able to visualize the paths taken by historical typhoons and the corresponding damages caused in USD with the year, prefecture, and categories as the main identifiers. Next, users can view the asset values of individual prefectures as well as the overall asset value spread across Japan. In addition, the visualisation of environmental factors and typhoon counts with the addition or removal of prefectures allows the identification of plausible trends. Lastly, users can visualise the simulated typhoon path and the projected damage caused upon selection of coordinates on the map *(Figure 15).*

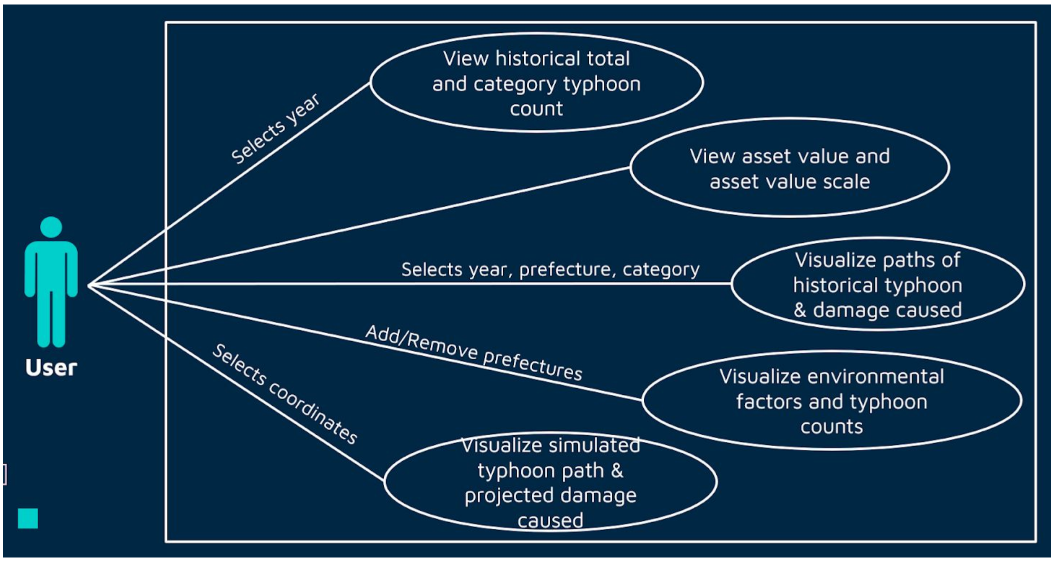


Figure 15: Use case diagram

3.4 User Scenarios Diagram

View historical typhoons count based on category by selecting the years

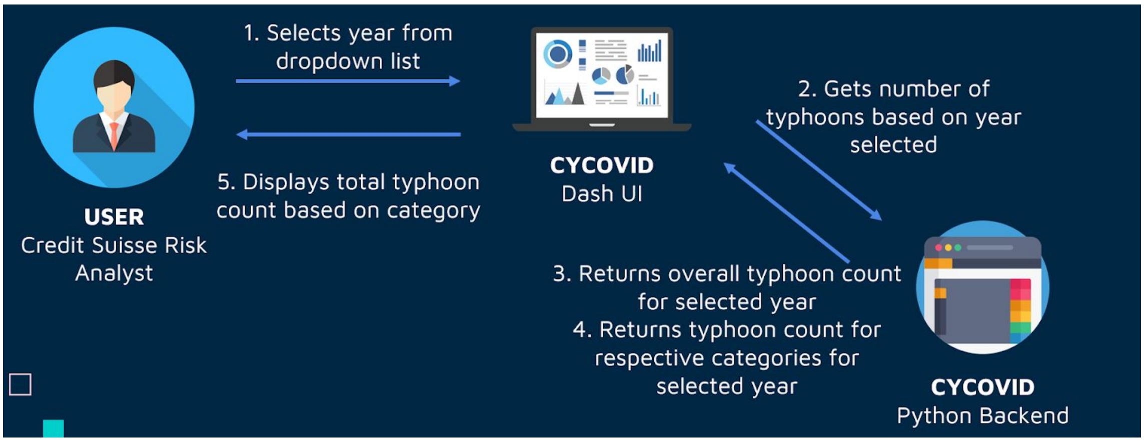


Figure 16: User scenario 1

View historical typhoon paths and the damages incurred

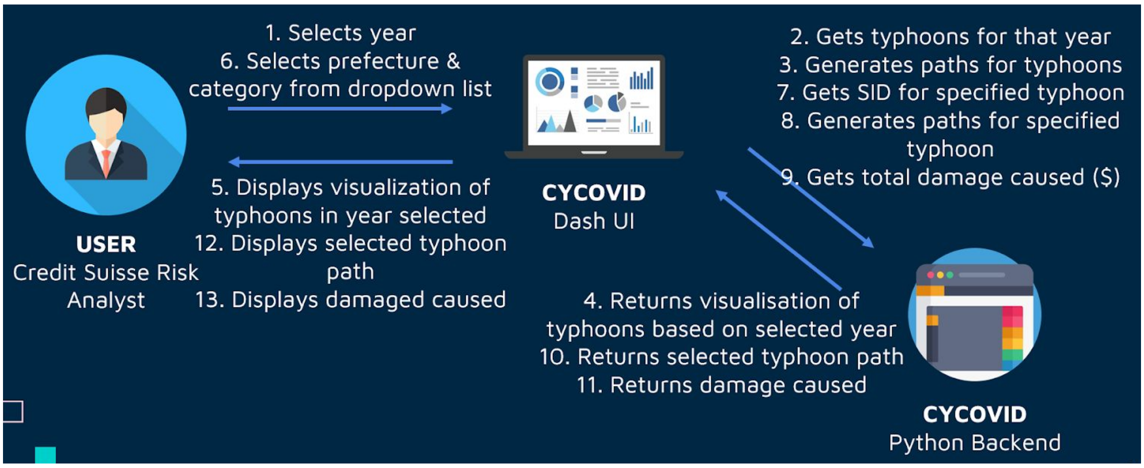


Figure 17: User scenario 2

Visualise asset spread and asset value

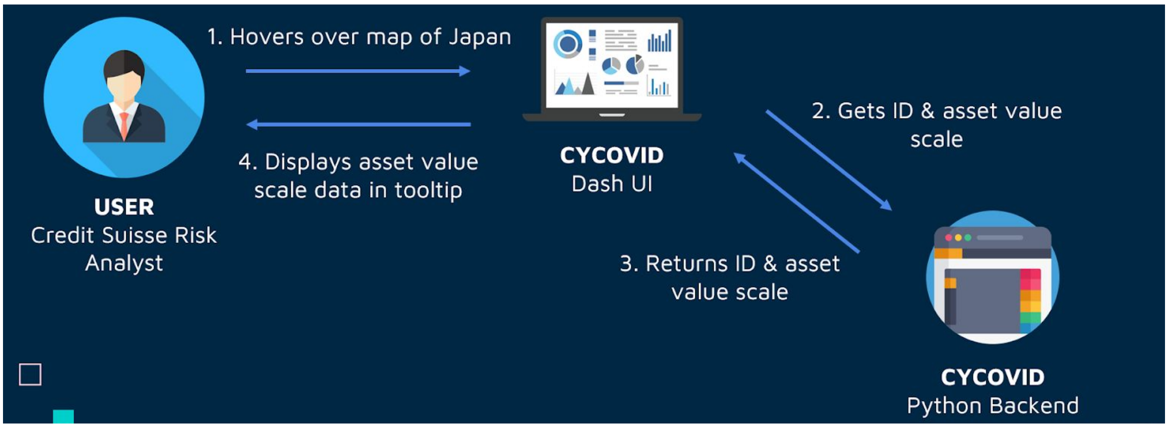


Figure 18: User scenario 3

Visualise environmental factors and typhoon counts

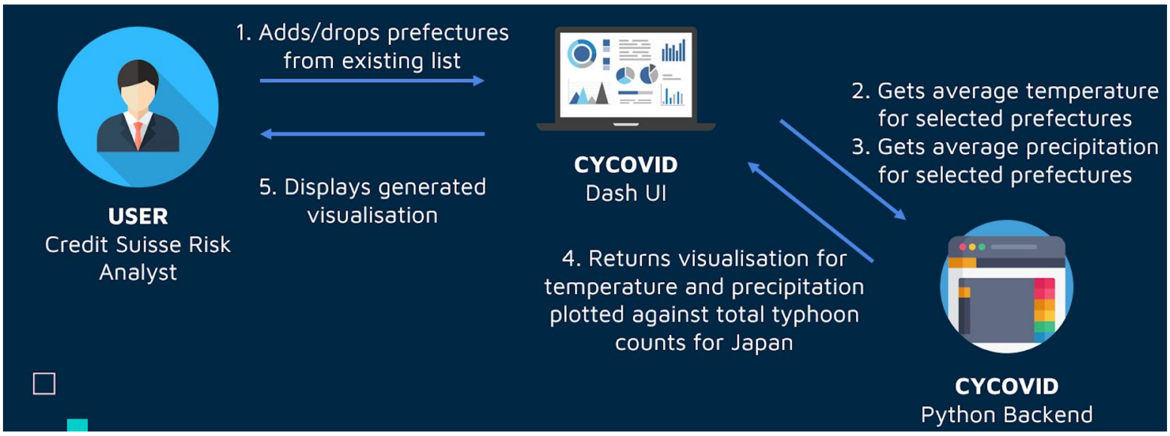


Figure 19: User scenario 4

Visualise simulated typhoon path and projected damage caused

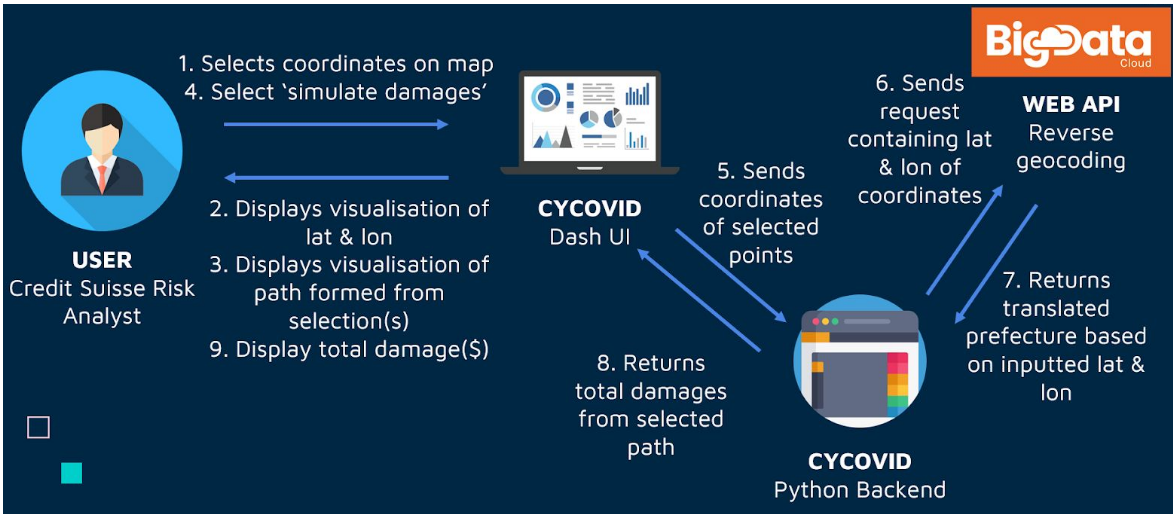


Figure 20: User scenario 5

3.5 Solution Model Comparison

The following *(Figure 21, 22, 23)* shows the before and after screenshots of the application. On the left would be the before and on the right would be after the improvements the team has made on the application since the midterm.

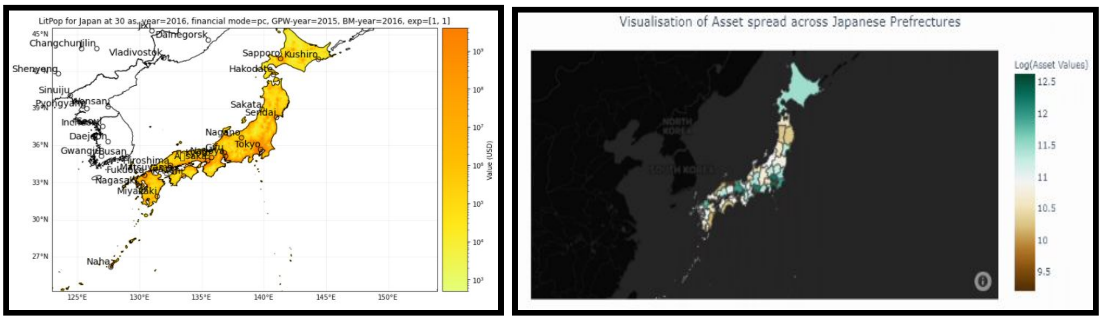


Figure 21: Asset spread across Japan

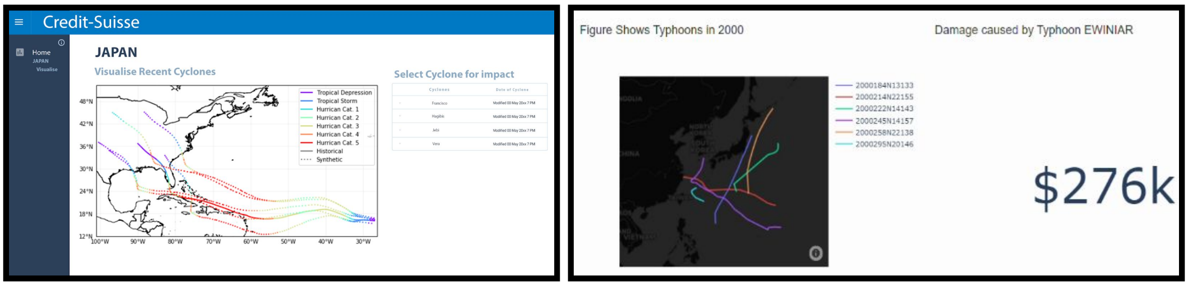


Figure 22: Historical paths of typhoons and damage incurred

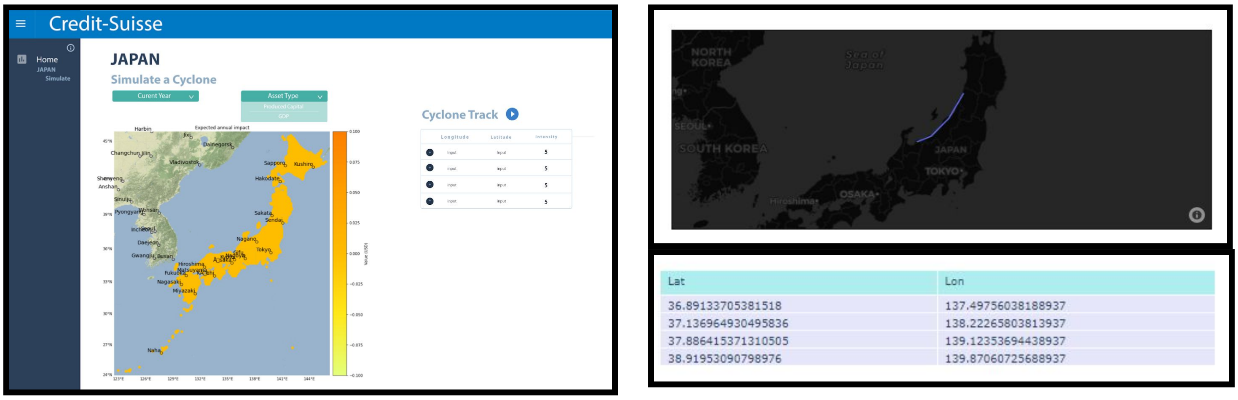


Figure 23: Simulated paths of typhoons

4. Technical Details

4.1 Software Tools & Framework

Programming Languages	
Python Programming language	Python was the main language used in the application. Python was not just used in data cleaning but also in the application's backend logic.
CSS	The team used a custom CSS file to configure the layout for the dash application.
R Programming Language	Since a similar data cleaning code was written by the team members in R language prior, the team made use of the code to reverse geocode coordinates during the data collection phase.
Database	
CSV files	For ease of working with the dataset, the team used locally stored CSV files. The information in these files can be transferred over to SQL or other databases once

	finalised.
Software & Tools	
GitHub Shared Directory	The application directory was pushed onto the team's private Git repository. From here, the team uses Github to collaborate on the application.
Google Drive Administrative Directory	For Documents such as weekly meeting minutes, collaborative report writing as well as presentation slides creation, the team used GoogleDocs as our main administrative platform.
Club House Project Management Tool	Club House has a simple and user friendly interface which allows the team to collaborate seamlessly and visually see personal deadlines that we were required to meet.
Frameworks & Libraries	
CLIMADA Economics of Climate Adaptation Open-sourced natural disaster impact simulator	The team used Climada's algorithm to generate asset value data for our project. Such data is not made available to the public.
plotly | Dash Dash framework for the User interface + Plotly.JS python compatible library for visualisations	The dash framework is the foundation of the application. Written on top of flask, plotly.js and react.js, it allows the team to create html components and js visualisations through writing in python language. The team used the library to build the web application as well as build plotly and leaflet visualisations. The plotly library was required for the team to visualise the data. Choropleth maps, indicators, maps and all visualisations in the the application were created using plotly with the exception of a leaflet map.

Leaflet Map + GoogleMaps Overlay	For the Simulation portion of the application, the team required a function that allowed us to retrieve Longitude and Latitude information when users clicked a point on the map. This was a difficult task as it was one of the limitations of writing plotly code in python. To work around this, the team realised that Leaflet had the functionality of google maps overlay and google maps itself had a built-in function to return longitude and latitude values upon click. Hence, the team used this method to develop the simulator in the web application. The library operated seamlessly with the dash application alongside with plotly.
Additional Software	
Telegram	The team used Telegram as our main communication platform. Short discussions, planning of meeting time and sharing of documents took place here. Sometimes the team also used it to have audio discussions when it was not necessary for us to share screens on Teams.
Microsoft Teams	The team created a group on MS Teams. This allowed us the convenience of meeting without needing to create a new room each time the team conducted a meeting. Anytime a member or the entire team needed a discussion, this would be the go-to platform for the team.
Skype For Business	Skype for Business was our Client's preferred method of communication. Hence, the team used this platform for our weekly meetings with the client.
Adobe Premiere Pro	This was the software the team used for the creation of our project video. It allied us with the ease of trimming and overlaying audio.

4.2 Application Architecture

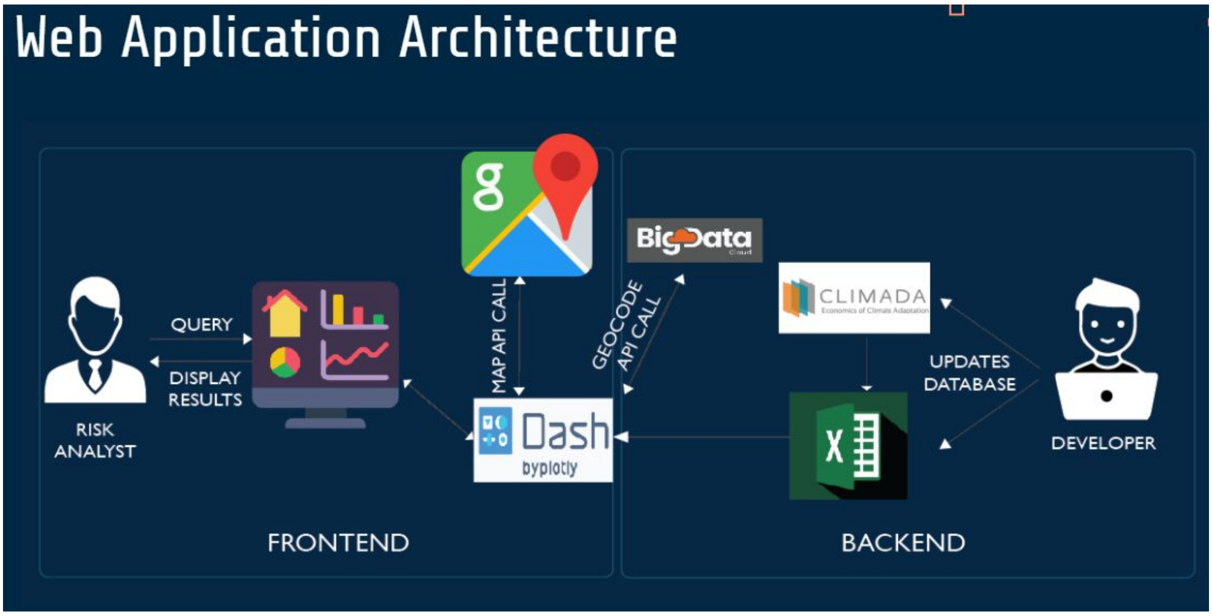


Figure 24: Web application architecture

4.2.1 Backend

Backend will mainly be data files updated by the developer. Due to the nature of typhoons, it is difficult to update information in real-time. Typhoon paths as well as climate changes of the current year should be updated annually when relevant data is being released. While asset data information can be updated every 5 years as the population data required by Climada's algorithm gets updated every 5 years as well (more information is provided in data collection - Algorithm generated dataset). With regular updates at these intervals, the team ensures that the application is capable of providing the most accurate results.

4.2.2 Frontend

Using dash, the team is able to write the User Interface in Python. This gave us the flexibility to integrate backend logic with front end code as well. Prior to visualisation on the web application, certain API calls are made by the application to google maps and Big Data Cloud. Google Maps API is required for the team to integrate retrieval of longitude and latitude through an overlay while Big Data Cloud is required for reverse-geocoding of coordinates for simulation.

All the interactions and visualisations the user observes are written in python, specifically the plotly and leaflet.js library.

4.3 Data Collection, Cleaning and challenges (Application)

Due to the nature of the project, data was acquired from public domains. Some of which the team was recommended by the client while other information was acquired through scrapping as well as processing via complex python algorithms.

4.3.1 Client Recommended Dataset

Historical Cyclone data -IBTrACS
https://www.ncdc.noaa.gov/ibtracs/index.php?name=ib-v4-access

Open-source CSV file containing information of typhoon path, longitudes, latitudes, typhoon category, name, etc for global typhoons from 1842 to the present date *(Figure 25)*.

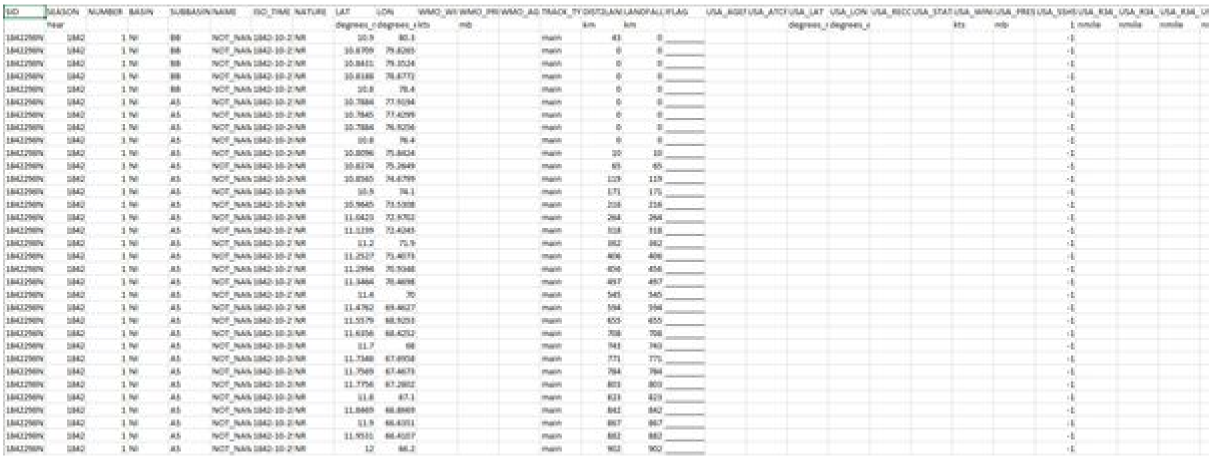


Figure 25: Original IBTrACS dataset

Cleaning Process	
Deriving Additional Geographic Information	Using the reverse_geocoder python library (https://pypi.org/project/reverse_geocoder/), the team added new columns to determine country as well as the specific geographic prefecture that each coordinate of a typhoon path belonged to.
Removing unnecessary information	Cleaning the information for this data required the team to remove unused columns such as empty fields, US-specific metrics, wind directions. After the Selection of Japan for the country-specific application, the team removed typhoons that belonged to other countries as well. Naming conventions of the remaining columns were altered to make it user friendly *(Figure 26).*

	A	B	C	D	E	F	G	H	I	J	K
1		SID	SEASON	NAME	ISO_TIME	LAT	LON	CATEGORY	COUNTRY	PREF	
2	0	2000184N:	2000	KIROGI	3/7/2000 18:00	17.05	131.65	1	PH	Eastern Visayas	
3	1	2000184N:	2000	KIROGI	3/7/2000 21:00	17.2488	131.661	1	PH	Eastern Visayas	
4	2	2000184N:	2000	KIROGI	4/7/2000 0:00	17.5	131.675	1	PH	Eastern Visayas	
5	3	2000184N:	2000	KIROGI	4/7/2000 3:00	17.87	131.668	2	PH	Bicol	
6	4	2000184N:	2000	KIROGI	4/7/2000 6:00	18.3	131.65	3	PH	Bicol	
7	5	2000184N:	2000	KIROGI	4/7/2000 9:00	18.7231	131.627	3	JP	Okinawa	
8	6	2000184N:	2000	KIROGI	4/7/2000 12:00	19.15	131.6	4	JP	Okinawa	
9	7	2000184N:	2000	KIROGI	4/7/2000 15:00	19.57	131.538	4	JP	Okinawa	
10	8	2000184N:	2000	KIROGI	4/7/2000 18:00	19.975	131.55	4	JP	Okinawa	
11	9	2000184N:	2000	KIROGI	4/7/2000 21:00	20.3407	131.732	4	JP	Okinawa	
12	10	2000184N:	2000	KIROGI	5/7/2000 0:00	20.725	132	4	JP	Okinawa	
13	11	2000184N:	2000	KIROGI	5/7/2000 3:00	21.2	132.257	4	JP	Okinawa	
14	12	2000184N:	2000	KIROGI	5/7/2000 6:00	21.675	132.525	4	JP	Okinawa	
15	13	2000184N:	2000	KIROGI	5/7/2000 9:00	22.0593	132.786	3	JP	Okinawa	
16	14	2000184N:	2000	KIROGI	5/7/2000 12:00	22.425	133.05	3	JP	Okinawa	
17	15	2000184N:	2000	KIROGI	5/7/2000 15:00	22.8262	133.318	3	JP	Okinawa	
18	16	2000184N:	2000	KIROGI	5/7/2000 18:00	23.25	133.6	3	JP	Okinawa	
19	17	2000184N:	2000	KIROGI	5/7/2000 21:00	23.7075	133.918	2	JP	Kagoshima	
20	18	2000184N:	2000	KIROGI	6/7/2000 0:00	24.15	134.225	2	JP	Kagoshima	
21	19	2000184N:	2000	KIROGI	6/7/2000 3:00	24.5161	134.459	2	JP	Kagoshima	
22	20	2000184N:	2000	KIROGI	6/7/2000 6:00	24.875	134.7	2	JP	Kagoshima	
23	21	2000184N:	2000	KIROGI	6/7/2000 9:00	25.2701	135.026	1	JP	Kagoshima	
24	22	2000184N:	2000	KIROGI	6/7/2000 12:00	25.725	135.375	1	JP	Kagoshima	
25	23	2000184N:	2000	KIROGI	6/7/2000 15:00	26.2574	135.684	1	JP	Kagoshima	
26	24	2000184N:	2000	KIROGI	6/7/2000 18:00	26.85	136	1	JP	Kagoshima	
27	25	2000184N:	2000	KIROGI	6/7/2000 21:00	27.4261	136.341	1	JP	Kochi	
28	26	2000184N:	2000	KIROGI	7/7/2000 0:00	28.1	136.725	1	JP	Wakayama	
29	27	2000184N:	2000	KIROGI	7/7/2000 3:00	28.9368	137.161	1	JP	Wakayama	
30	28	2000184N:	2000	KIROGI	7/7/2000 6:00	29.875	137.65	1	JP	Wakayama	
31	29	2000184N:	2000	KIROGI	7/7/2000 9:00	30.8112	138.175	1	JP	Wakayama	
32	30	2000184N:	2000	KIROGI	7/7/2000 12:00	31.825	138.775	1	JP	Shizuoka	
33	31	2000184N:	2000	KIROGI	7/7/2000 15:00	32.9541	139.322	1	JP	Shizuoka	
34	32	2000184N:	2000	KIROGI	7/7/2000 18:00	34.125	139.9	1	JP	Chiba	
35	33	2000184N:	2000	KIROGI	7/7/2000 21:00	35.2793	140.66	1	JP	Chiba	

Figure 26: Example of cleaned dataset

Challenges	
Long time taken for reverse_geocoder python Library	While the library provided accurate information, it was not quick to run. The team attempted Google Maps API as well but due to the sheer volume of data the team needed to reverse geocode, it would be a very costly task. Hence, the team focused only on cyclones & typhoons from the most recent 20 years; 2000-2020. This reduced the workload on the library allowing it to clean the data in 2 hours.
Understanding the IBTRACS data	Since IBTRACS contained many naming conventions, the team was unsure what each data represented and which were the data the team needed to use. It took some time for the team to understand the column documentation that was provided for us at the same source as IBTRACS data.

4.3.2 Self Collected Dataset

GeoJson - https://tmfraser65.carto.com/tables/japan_prefectures/public/map

For the visualisation of data such as the spread of assets or even risk differentials across prefectures, the team thought that the use of a choropleth map *(Figure 27)* would be the best fit. Hence, the team found appropriate geojson data represented in terms of prefectures. This data required no cleaning.

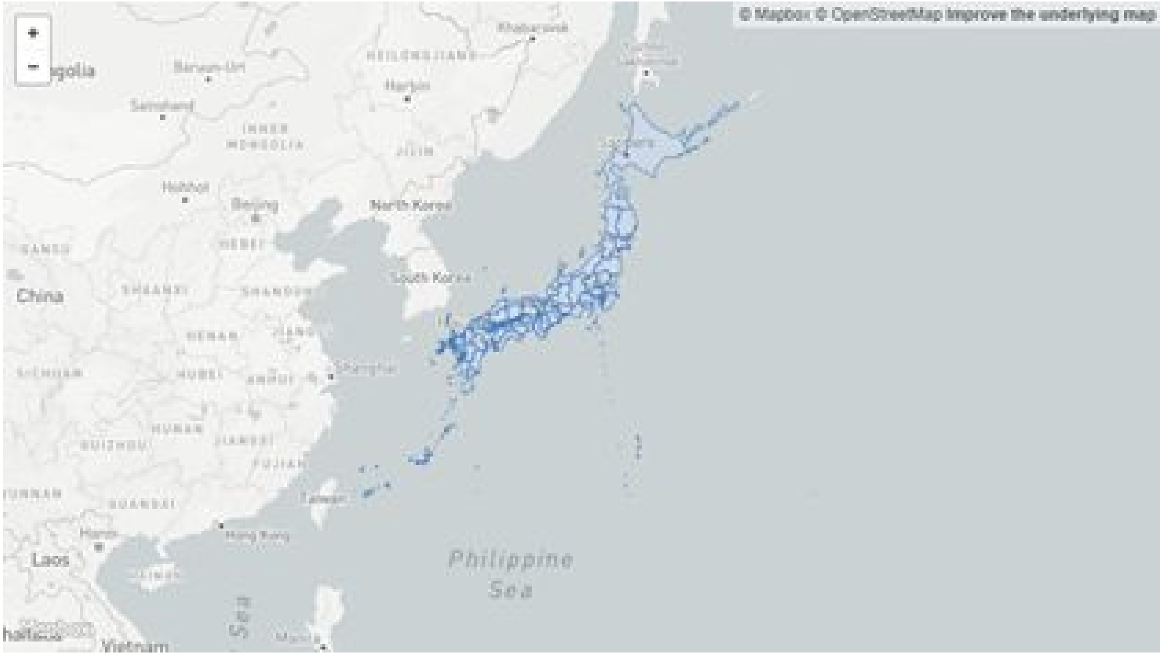


Figure 27: Example of a choropleth map conjured from GeoJson information

Climate data - Japan Meteorological Agency

Since Japan faces different climates depending on the area of the country an individual is in, it was difficult to collect data that gave us a general overview of Japan. The Meteorological Agency provides data captured at various weather stations throughout Japan from the 1800s to the present date. For the use in the application, the team took annual averages of temperature and precipitation.

Challenges	
Manual data collection	Because the data was not large enough for us to use a web scraper (since it would take more time to write the script), the team manually collected information from 10 prominent prefectures in Japan.
Representation of Information	Some prefectures are not represented by weather stations. Hence, to have an overview of Japan's information, the team had to take a sample of 10 prefectures as information on all the prefectures were unavailable.

4.3.3 Algorithm Generated Data

Through the qualitative research process, the team was unable to find the necessary information required to provide asset exposure information required for the risk analysis of

Japan. Such data was either sensitive or not well documented enough to be collected. To top it off, financial jargons such as produced capital came with various different definitions depending on which financial institute or country was providing the data. Since the team wanted to have a proper base of comparison, it became imperative that we needed to simulate the data.

<table>
<tr><th colspan="2">Cleaning Process</th></tr>
<tr><td>Algorithm Used</td><td>The team will be taking the asset distribution over a grid proportional to Lit^m x Pop^n on a map. Lit represents the amount of night light while pop represents the population size of that particular area in Japan.
For this algorithm to work, the team required global population data. The team collected this information from https://sedac.ciesin.columbia.edu/data/collection/gpw-v4/sets/browse
which provided the population density of the world according to grids. This data is updated every 5 years; currently, data includes 2000, 2005, 2010,..., 2020.
The team then used the Litpop function of climada to generate the spread of assets across Japan using data that Climada collects from Credit Suisse and World Bank DataBank.</td></tr>
<tr><td>Turning Coordinates into prefectures</td><td>

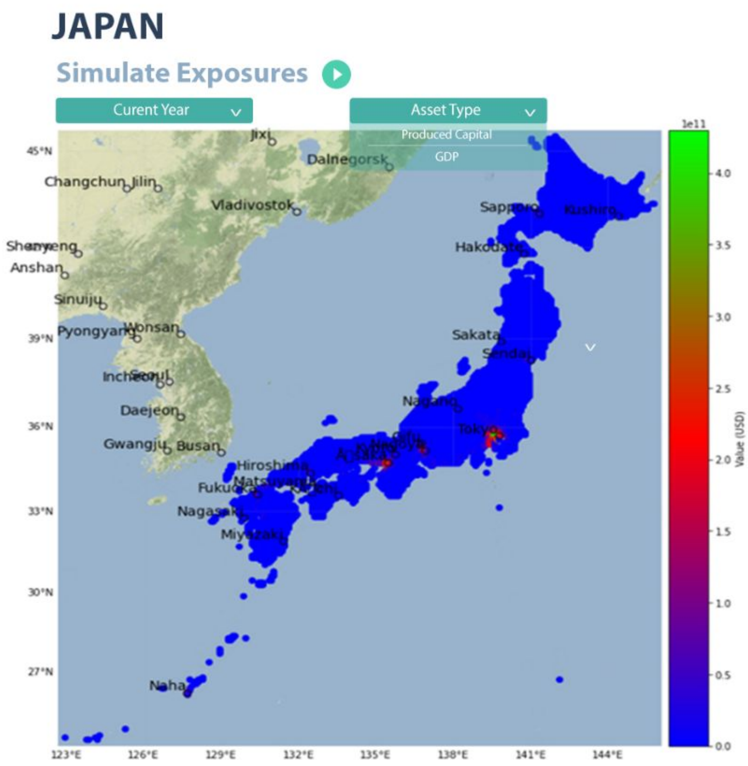


Figure 28: Visualisation of the production value
The generated data shows 3 outputs: Longitude, Latitude, Asset Value(USD). This has made it difficult for the team to</td></tr>
</table>

	visualise the data since the team needed to have prefecture information to work with a choropleth map. The team used R language and an open-source API: https://www.bigdatacloud.com/geocoding-apis/free-reverse-geocode-to-city-api?gclid=CjwKCAiAnIT9BRAmEiwANaoE1XxCkTRJ--tzS8_0-N_IxlSizYZrjdi7x8_d7qXhtVMy4YHfMh1n1hoCsa0QAvD_BwE to reverse geocode the coordinates into prefectures.

Challenges	Null Asset values resulting from the API calls. Due to naming conventions, certain prefectures became unrepresented and there were no values for them. The following are examples and how the team overcame these challenges.
Inconsistent Json formats	(see Figure 29 below)


```
{
  "latitude": 33.30780029296875,
  "longitude": 132.0,
  "plusCode": "8Q5J8252+42",
  "localityLanguageRequested": "en",
  "continent": "Asia",
  "continentCode": "AS",
  "countryName": "Japan",
  "countryCode": "JP",
  "principalSubdivision": "Iyo Province",
  "principalSubdivisionCode": "",
  "city": "",
  "locality": "Ikata",
  "postcode": "",
  "localityInfo": {
    "administrative": [
      {
        "order": 2,
        "adminLevel": 2,
        "name": "Japan",
        "description": "sovereign state in East Asia, situated on an archipela
        "isoName": "Japan",
        "isoCode": "JP",
        "wikidataId": "Q17",
        "geonameId": 1861060
      },
      {
        "order": 5,
        "adminLevel": 4,
        "name": "Ehime Prefecture",
        "description": "prefecture of Japan",
        "isoName": "Ehime",
        "isoCode": "JP-38",
        "wikidataId": "Q123376",
        "geonameId": 1864226
      },
      {
        "order": 6,
        "adminLevel": 4,
        "name": "Iyo Province",
        "description": "former province of Japan",
        "wikidataId": "Q907829"
      },
      {
        "order": 7,
        "adminLevel": 6,
        "name": "Nishiuwa district",
        "description": "district in Ehime prefecture, Japan",
        "wikidataId": "Q1206492",
        "geonameId": 1855138
      },
```


Figure 29: Json output from API call to Big Data Cloud

<table>
<tr>
<td></td>
<td>

The most accurate naming convention comes from order 5 with admin level 4 *(Figure 29)*. This will return us the exact prefecture name.

However, not all queries contained order 5 + admin level 4. Hence, the team would query the principal subdivision instead. This issue was mainly isolated to Saga Prefecture where certain locations had subdivisions “Principal Subdivision: Saga Prefecture” but did not have order 5+ admin level 4 to query from.

Hence, the team wrote some lines of python code.
E.g. pseudo code
If (order 5 + admin level 4) is not true:
    Prefecture = principal subdivision
    If Prefecture = “Saga Prefecture”:
        Prefecture = “Saga”

</td>
</tr>
<tr>
<td>Naming Conventions</td>
<td>

```
for rows in correct_df:
    prefecture = rows[1]
    value = rows[0]
    for prefectures in new_prefecs:
        counter = 0
        if prefecture == "Hyogo":
            if prefectures[0] == 'Hyōgo':
                prefectures[1] += value
            counter +=1
        if prefecture == "Nagasaki":
            if prefectures[0] == 'Naoasaki':
                prefectures[1] += value
            counter +=1
        elif prefecture == prefectures[0]:
            prefectures[1] += value
            counter +=1
```

Figure 30: Screenshot of a conversion function

Due to naming conventions, the API returned certain prefectures using the old naming convention such as in Nagasaki. Hyogo was also returned from the API without accents which made the data unrecognisable. Hence, the team had to do some conversion (*Figure 30)* such that the asset value data could be mapped to our existing prefecture data.

</td>
</tr>
<tr>
<td>Adding up the assets</td>
<td>

With the data cleaned, this became pretty straightforward. The team used a for loop and added the sum of each unique prefectures to create a list of prefectures and their corresponding asset value as such *(Figure 31)*:

</td>
</tr>
</table>

```
[['Iwate', 19426570296.47034],
 ['Kagawa', 74055028161.14862],
 ['Shimane', 12819636258.248997],
 ['Shizuoka', 235959175854.08508],
 ['Tokushima', 61229330686.10107],
 ['Tokyo', 4253867888342.6123],
 ['Nagano', 50418117082.870514],
 ['Niigata', 79561261050.90514],
 ['Oita', 30193301671.819103],
 ['Okayama', 112044484470.07597],
 ['Okinawa', 213686811234.5018],
 ['Osaka', 2430494290566.4],
 ['Saga', 60659840116.69025],
 ['Saitama', 1716056251648.947],
 ['Tottori', 1585585226.6509006],
 ['Toyama', 31187639194.124992],
```

Figure 31: List of prefectures with corresponding asset value

4.4 Technical Challenges

4.4.1 Framework Unfamiliarity

The team had proposed several shifts in the overall solution architecture throughout the duration of the project; with multiple jumps from Django, HTML + Vanillas Javascript, and Flask before finally settling down on implementing Plotly Dash with a local CSV backend.

4.4.2 Library Installations

The utilization of the external Climada library had led to various complications during library installations. The libraries in the makefile proffered incompatible versions, rendering different errors when running the library. Libraries had to be individually installed and updated to specific versions such that Climada was able to load.

4.4.3 Inaccurate API Calls

The formatting of API calls gave inconsistent Json outputs, leading to data inaccuracies. The prefectures have been wrongly labelled with old naming conventions, which had rendered the team's faulty results when placed into the application. An extra effort had to be placed into cleaning the data in order to achieve the correct tagging of prefectures.

4.4.4 Programme Incompetencies

Plotly was unable to take on click-inputs so the team had to find a solution through overlaying google maps to retrieve longitude and latitude values to aid the plotting of typhoon paths for the typhoon simulation functionality in the application.

4.4.5 Niche Python Libraries

The lack of documentation led to plenty of time spent on self-learning the concepts, resulting in a delay in the progress during the initial stages of the project.

5. User Acceptance Testing (UAT)

The team conducted two rounds of User Acceptance Testing. The target audience we have becas students and the general public. The reason why we have such demographics was that our client, Credit Suisse, wanted the application to be available for everyone. This includes the public that has zero background in typhoons to people who have an interest in putting their investments in Japan. The goals we have for UAT was to ensure that the application is usable from an end-user perspective as well as to identify and resolve any bottlenecks in completing the tasks. In addition, the application aims to provide both Credit Suisse and users the ability to visualise historical typhoons based on the categories of their interest, simulate typhoons based on user selection of coordinates on the map, viewing of damage reports (asset value), identifying prefectures with high frequencies of typhoons and last but not least to visualise typhoons occurrence in relation to the average temperature and precipitation annually. With all this information made available to the users, they will be able to make better-informed decisions before placing any investments in Japan.

5.1 UAT Goals

- User finds the application user-friendly and intuitive
- User is able to perform and complete all the tasks independently
- User is able to locate and understand how to simulate a typhoon
- User is able to interpret the asset damage table that was being generated

5.2 UAT Tasklist

The team has included the UAT document for the users inside the zipped submission *(Figure 32)*.

Task	
1	Locate Tokyo on the map. Write down its asset value scale.
2	Write down the asset value of Tottori.
3	Write down the latitude and longitude of Typhoon Choi-Wan (a category 3 typhoon) which took place in 2003.
4	Write down the total number of cyclones and number of Category 4 cyclones that took place in 2017 in Japan.
5	Write down the color representing the path of cyclone Bolaven in 2012.
6	Write down the average temperature for Tokyo and Osaka in 2020.
7	Write down the number of times Tokyo was hit by cyclones in 2015.
8	Simulate a cyclone of your choice and write down the predicted damage amount.
9	Overall feedback for the web appliaction.

Figure 32: UAT tasks

5.3 UAT Results & Insights

5.3.1 UAT Results & Insights for 1st Round

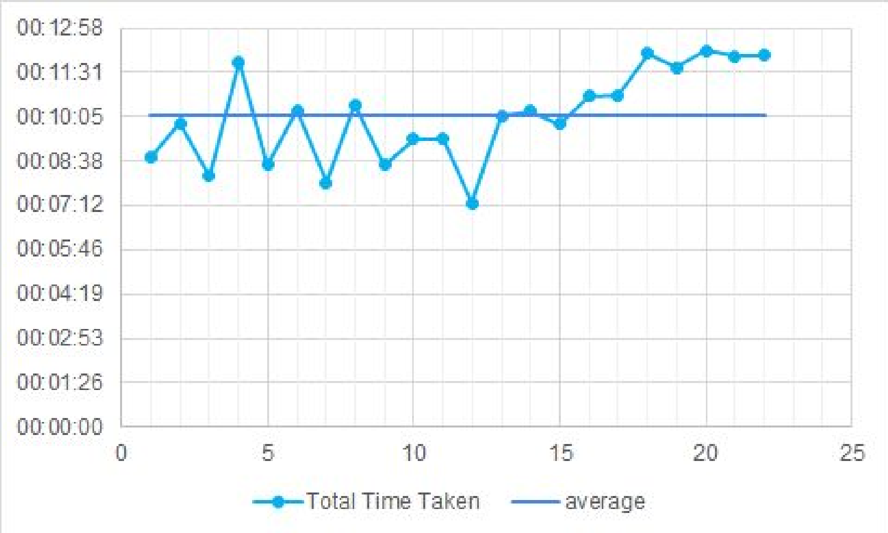


Figure 33: Time taken for 1st round of UAT results

	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8
	Time taken	Time taken	Time taken	Time taken	Time taken	Time taken	Time taken	Time taken
Task 1	00:00:42	00:00:53	00:01:03	00:02:07	00:00:54	00:01:36	00:01:05	00:00:45
Task 2	00:00:44	00:01:10	00:00:58	00:01:19	00:00:47	00:00:54	00:00:42	00:00:56
Task 3	00:00:49	00:02:08	00:01:34	00:02:30	00:01:52	00:01:48	00:01:21	00:01:48
Task 4	00:00:57	00:00:38	00:00:31	00:01:02	00:00:48	00:00:55	00:00:59	00:01:04
Task 5	00:00:38	00:01:01	00:00:49	00:00:37	00:00:31	00:00:58	00:00:34	00:00:47
Task 6	00:00:58	00:01:12	00:00:38	00:00:57	00:00:37	00:00:56	00:00:42	00:01:34
Task 7	00:01:42	00:00:54	00:00:34	00:00:36	00:00:43	00:02:09	00:00:31	00:00:54
Task 8	00:02:15	00:01:56	00:02:01	00:02:44	00:02:18	00:01:01	00:02:03	00:02:41
Total Time Taken	00:08:45	00:09:52	00:08:08	00:11:52	00:08:30	00:10:17	00:07:57	00:10:29
Average Time Taken for All Respondents to complete 8 tasks	00:10:06							
Wrong answer		7	6,7	6,7		6,8		

Figure 34: Time taken per task for 1st round of UAT results

The team conducted the first round of UAT with a total of 22 users *(Figure 34)*. Keeping in mind the importance of usability in the application, the team decided to limit the time taken for users to complete each task, setting a maximum of 3 minutes per question and they were to perform it independently without any aid from the team. Users will be asked to move on once 3 minutes are up. Based on the results, only 10 users made it to the end with all the correct answers. The average time taken for them to perform and complete all 8 tasks is approximately 10 minutes.

Insights Gathered from Users

Figure 35: 1st round word cloud

The team put up the feedback into a word cloud above (*Figure 35*) to identify some of the key pain points that users felt after they used the application. Some of the common words mentioned by them were “difficult”, confusing”, “cannot find”. The users in general find the application confusing and hard to use due to the absence of labels, titles, and segmentations of the visualisations.

5.3.2 Improvements Made

After filtering out all the improvements that participating users have suggested, the team has derived the top 3 priority changes the team aims to improve on (*Figure 36, 37, 38*).

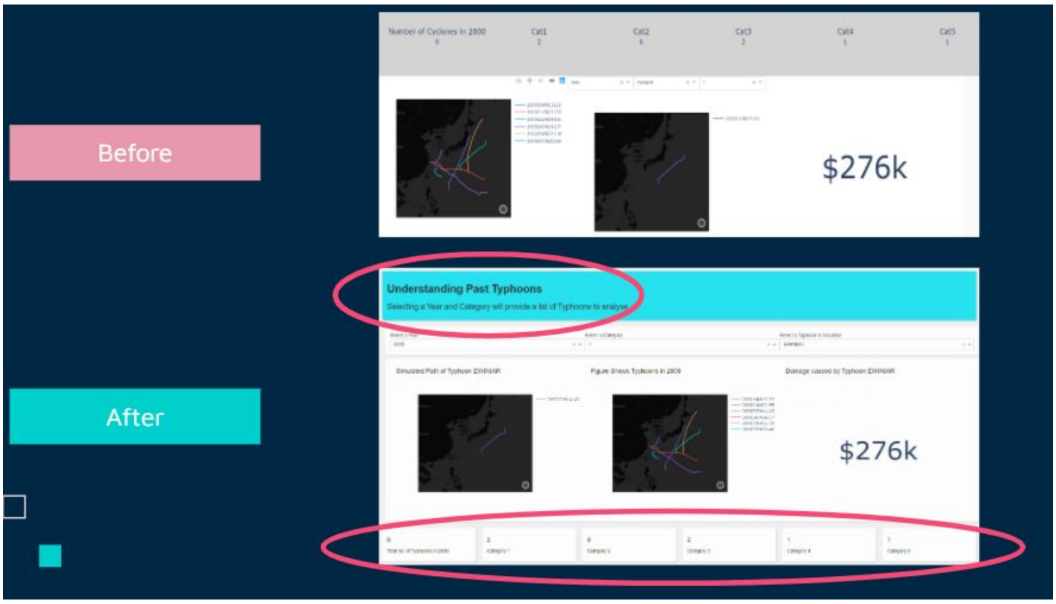


Figure 36: Adding labels and titles to all the visualisations

Figure 37: Having proper segmentations

Figure 38: Fixing Japan to boundaries of the simulation map to improve user-friendliness and better present the overall application

5.3.3 UAT Results & Insights for 2nd Round

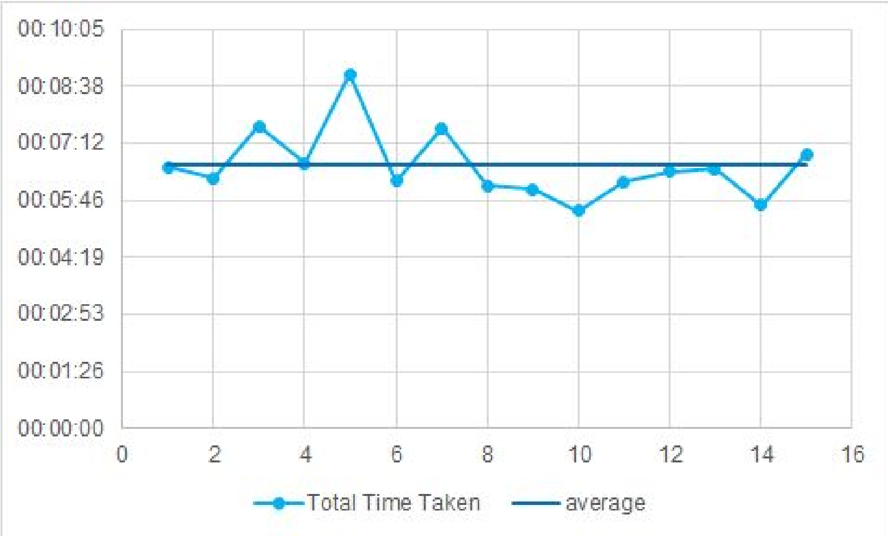


Figure 39: Time taken for 2nd round of UAT results

	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8
	Time taken	Time taken	Time taken	Time taken	Time taken	Time taken	Time taken	Time taken
Task 1	00:00:31	00:00:44	00:00:42	00:00:53	00:01:48	00:00:32	00:00:47	00:00:38
Task 2	00:00:38	00:00:39	00:00:57	00:00:40	00:01:19	00:00:47	00:00:41	00:00:38
Task 3	00:00:41	00:00:54	00:01:47	00:01:05	00:01:24	00:00:56	00:01:38	00:00:49
Task 4	00:00:39	00:00:45	00:00:32	00:00:33	00:00:40	00:00:41	00:00:48	00:00:43
Task 5	00:00:32	00:00:37	00:00:51	00:00:49	00:00:37	00:00:31	00:00:33	00:00:39
Task 6	00:00:51	00:00:58	00:00:59	00:00:38	00:00:57	00:00:46	00:00:39	00:00:41
Task 7	00:01:02	00:00:47	00:00:46	00:00:43	00:00:36	00:00:53	00:00:45	00:01:11
Task 8	00:01:42	00:00:55	00:01:04	00:01:21	00:01:35	00:01:08	00:01:44	00:00:48
Total Time Taken	00:06:36	00:06:19	00:07:38	00:06:42	00:08:56	00:06:14	00:07:35	00:06:07
Average Time Taken for All Respondents to complete 8 tasks	00:06:38							

Figure 40: Time taken per task for 2nd round of UAT results

With the improvements made to the application (the top 3 priority changes given by the users in the first round of UAT), the team conducted a second round of UAT and brought the application to 15 new users and had a perfect completion rate as seen on (*Figure 40)*. The average time taken to complete the tasks was 6.5 minutes, a significant improvement with the time taken greatly reduced by approximately 3.5 minutes.

<u>Insights Gathered from Users</u>

Figure 41: 2nd round word cloud

This time around, as seen on the word cloud above (*Figure 41*), the team received positive feedback from the users mentioning that the application was interesting and visualisations were clearly labelled with titles which helps them to easily locate relevant information that was needed. Overall, the users find the application intuitive and easy to use.

6. Final Product

6.1 User Interface

6.1.1 Analysis of Historical Typhoons

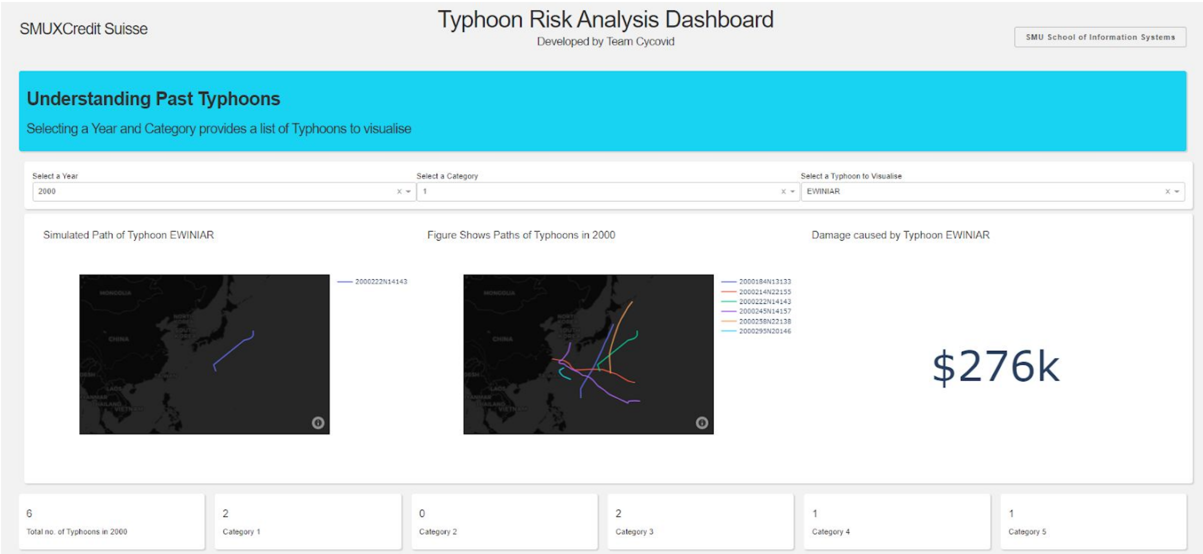


Figure 42: Historical typhoons chart

Users will be able to view paths of past typhoons in the last 20 years. Upon selection of a specific typhoon, the path, as well as exact damage incurred, will be shown.

6.1.2 Asset and Climate Overview

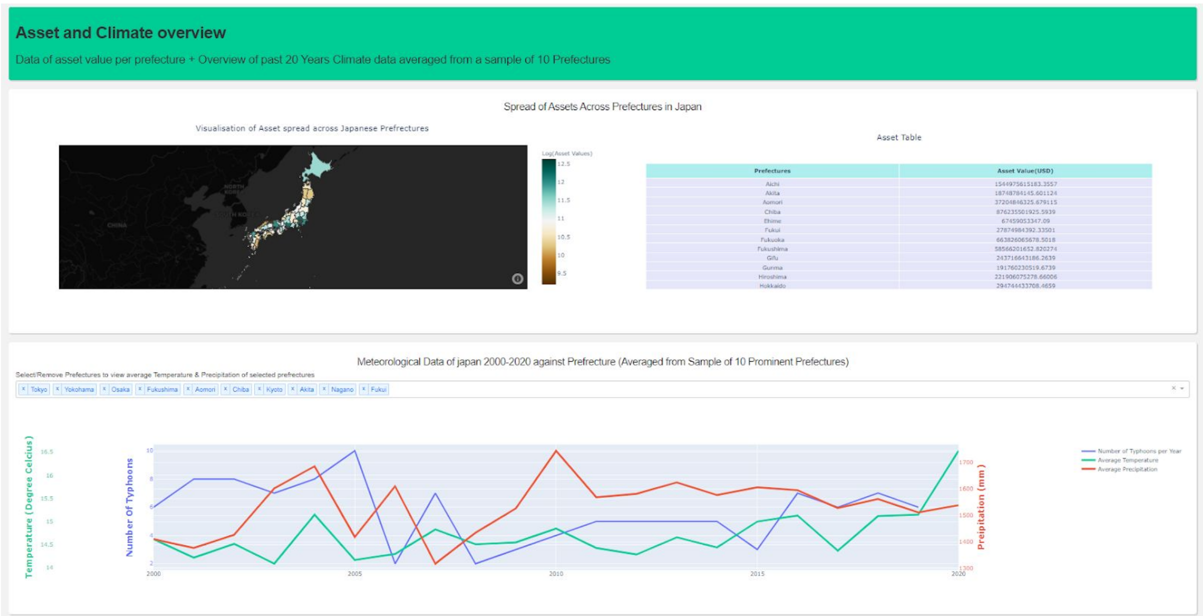


Figure 43: Asset and climate charts

The spread of assets across Japan is shown. Users are able to view a choropleth map and hover across prefectures to view their monetary value.

Users will be able to analyse the trend of average precipitation and average temperature to better understand its correlation with the number of tropical typhoons occurring.

6.1.3 Risk Analysis of Prefectures Across Japan

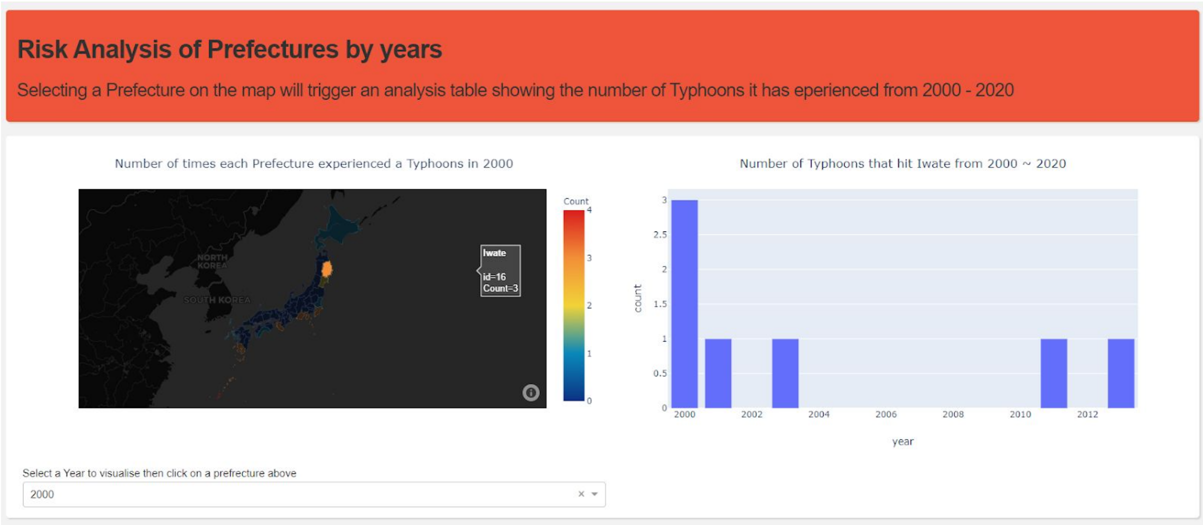


Figure 44: Risk analysis of prefectures across Japan chart

The users will be able to view the number of times a prefecture is hit by typhoons in a selected year to understand the prefecture's risk profile.

6.1.4 Typhoon Simulator

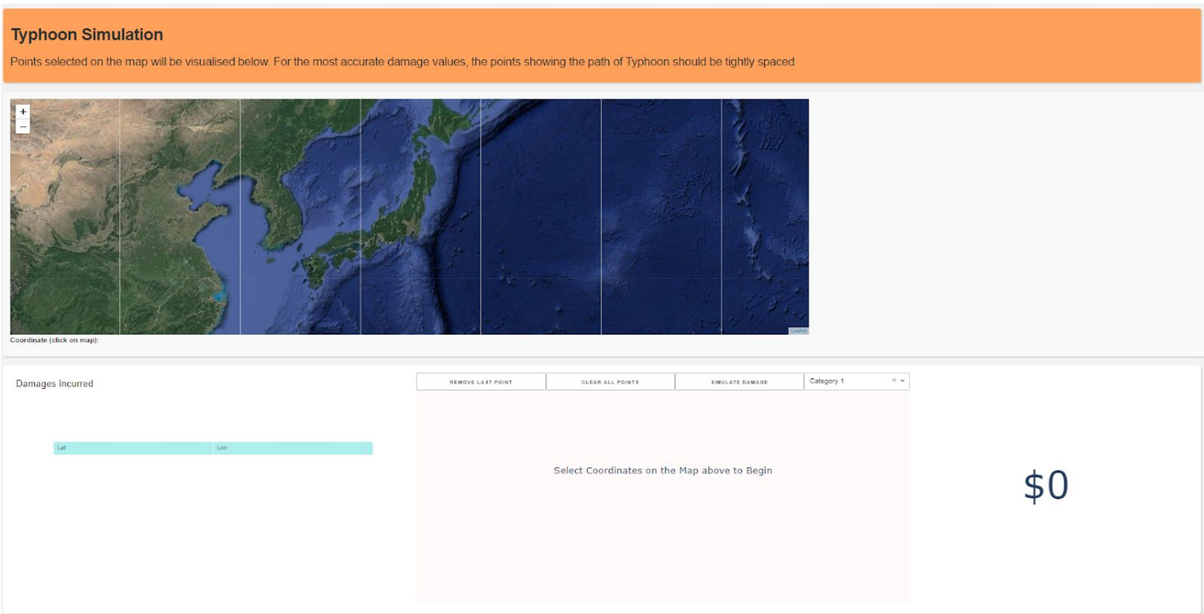


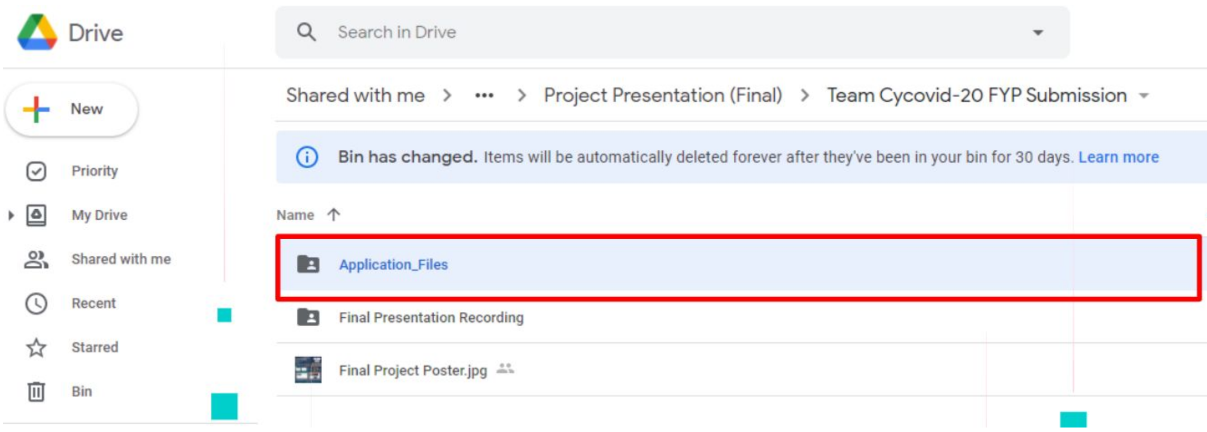
Figure 45: Typhoon simulation

This part of the application allows users to generate their own typhoon path and simulate damages for the path.

6.2 In-Depth User Guide

6.2.1 Installation Guide

Step 1: Installing the files locally



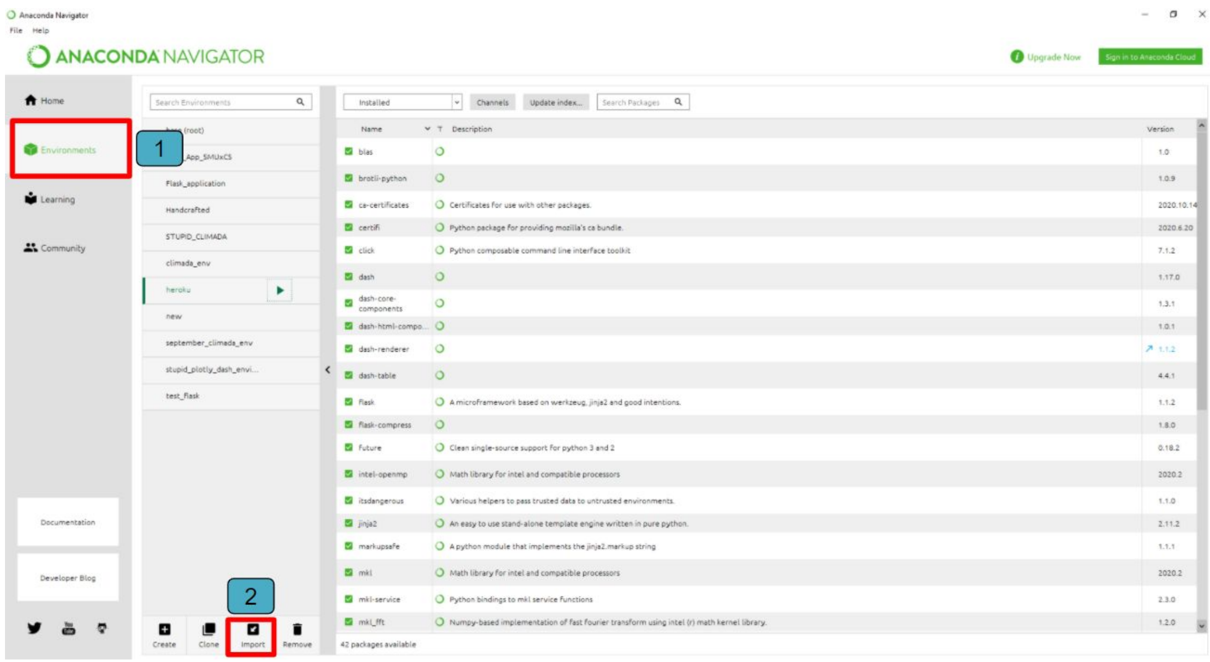
In the google docs link:
https://drive.google.com/drive/folders/1e6_IXjsInDkL5zszAK_diUxjLcMEZ65U?usp=sharing
Locate and install the folder called Application_Files.

Step 2: Install anaconda

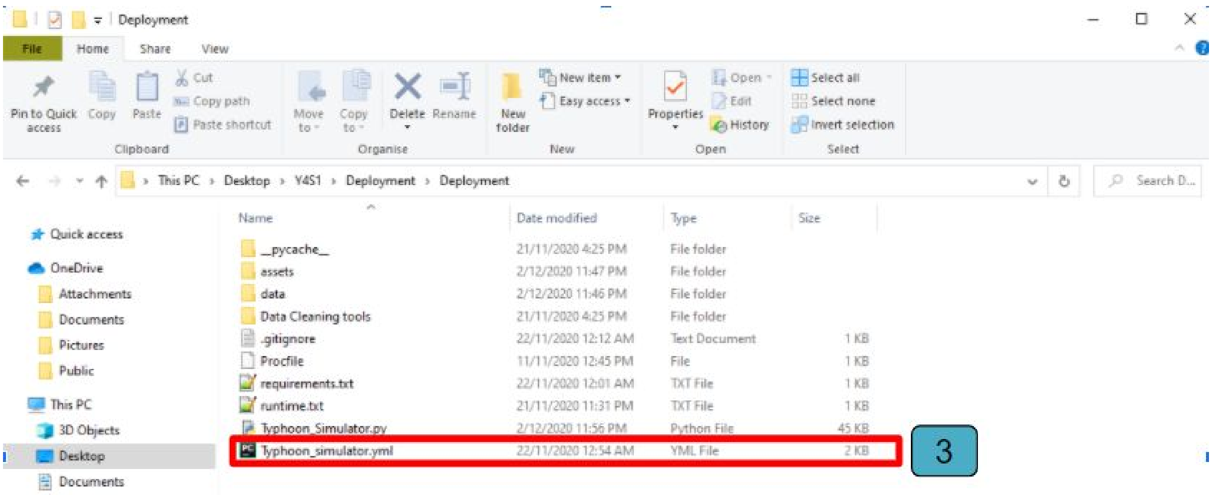
An installation guide is provided in the link below.
https://docs.anaconda.com/anaconda/install/windows/

Step 3: Install the packages required to run the application on Anaconda

Firstly, the user will launch the anaconda application and navigate to the environments.

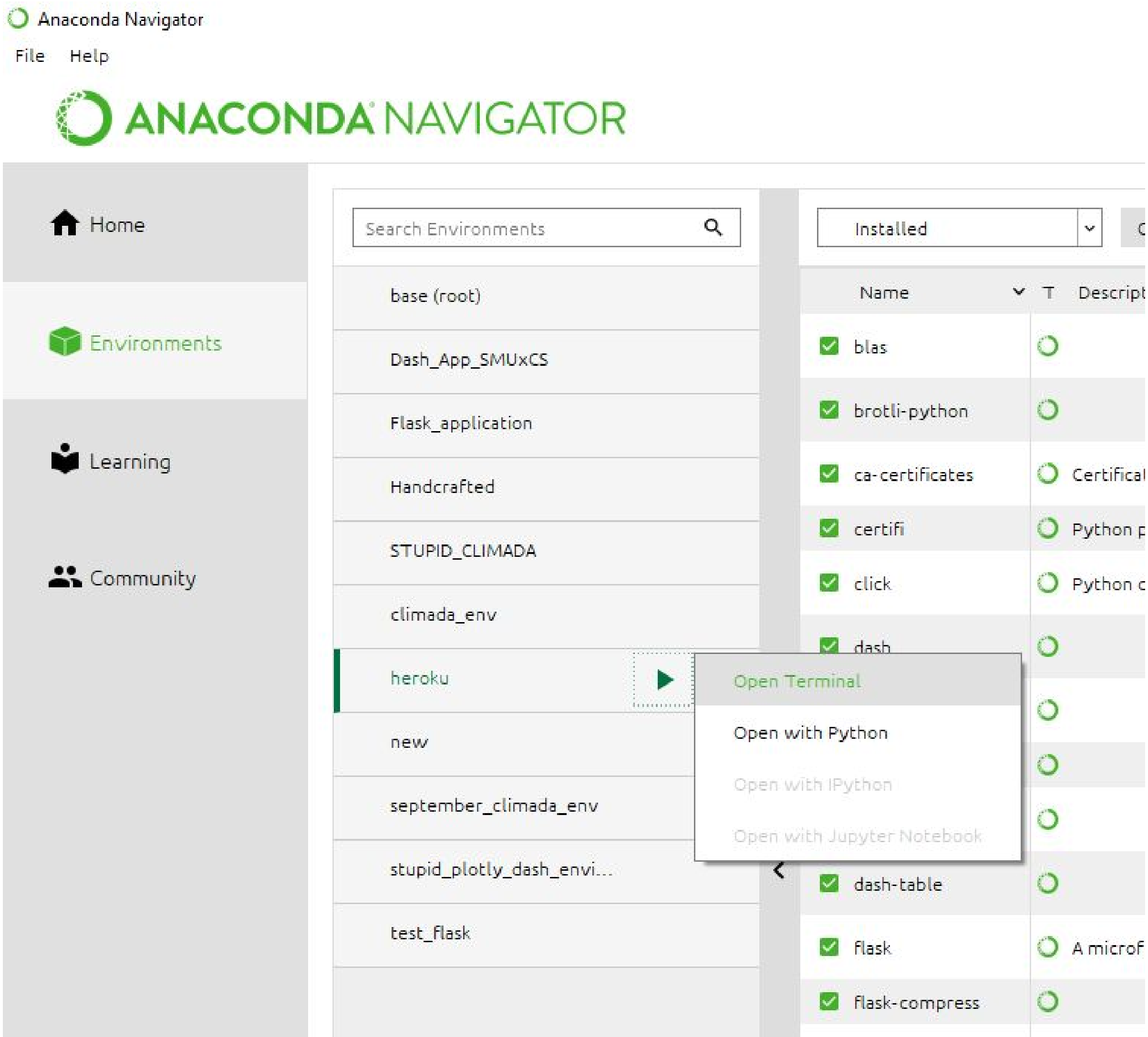


Clicking on environments in the side tab will bring the user to this page. After which, navigate to the Import button [2] and click it. This opens up the file directory where the user will navigate to the directory of the installed application folder.



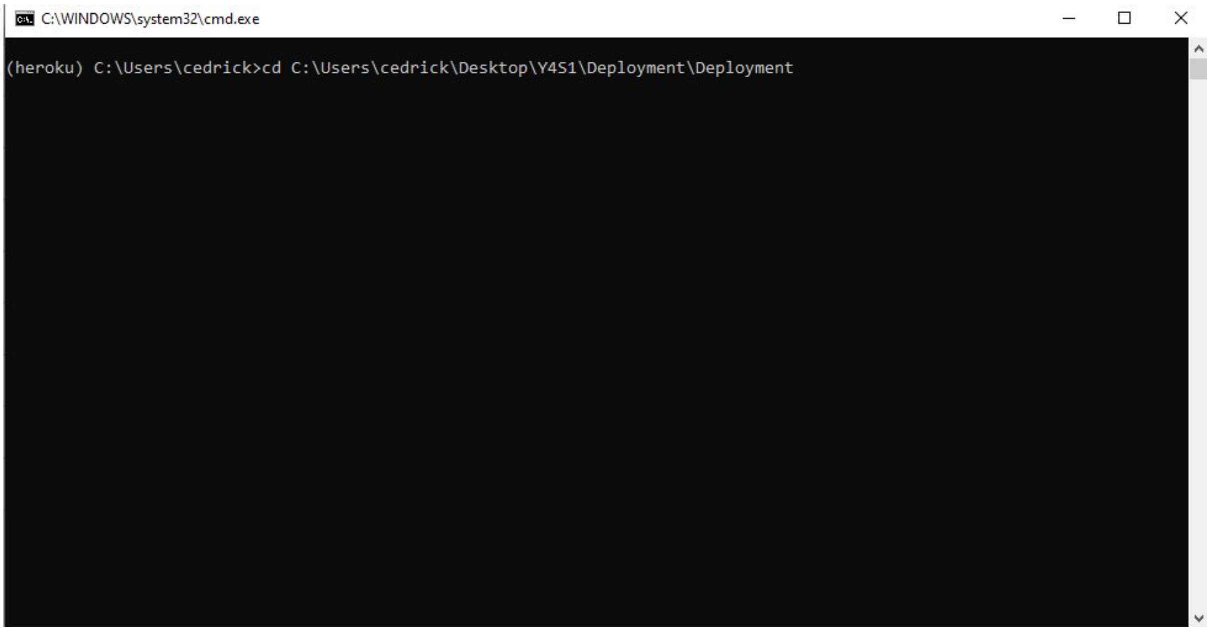
Locate the “Typhoon_Simulator.yml” file [3] and double click it. This will install all dependencies required to run the application.

Step 4: Launching the application

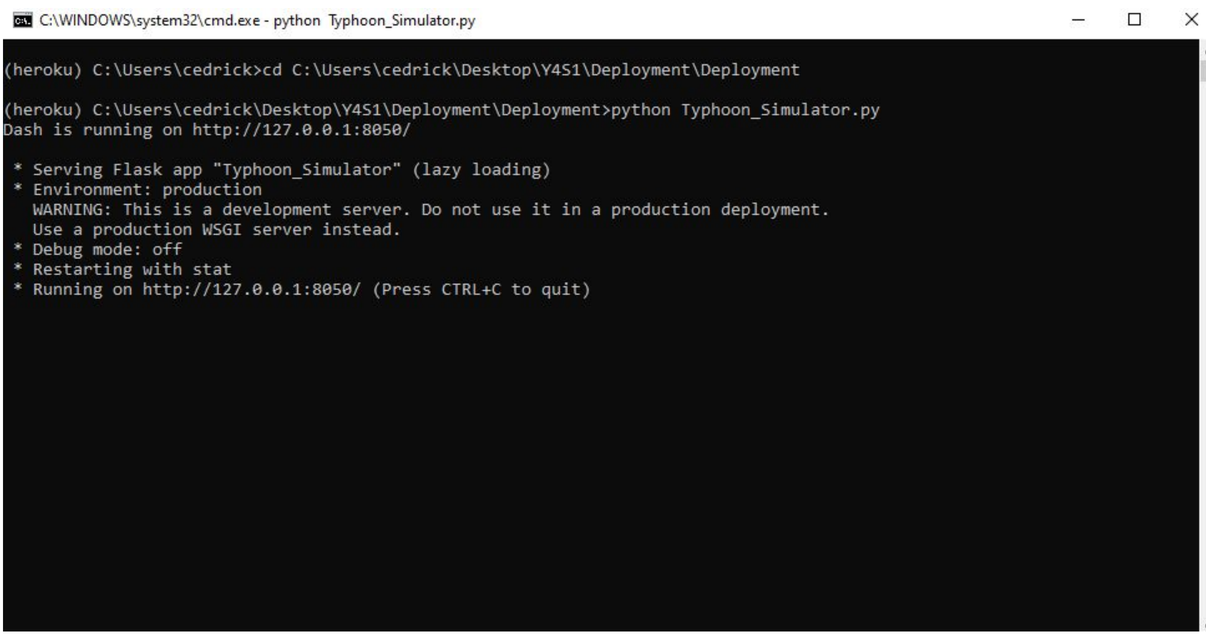


In anaconda, click the arrow and open the terminal on the newly installed Conda environment. Installed from the YML file, the environment should be named “Typhoon_Simulator”.

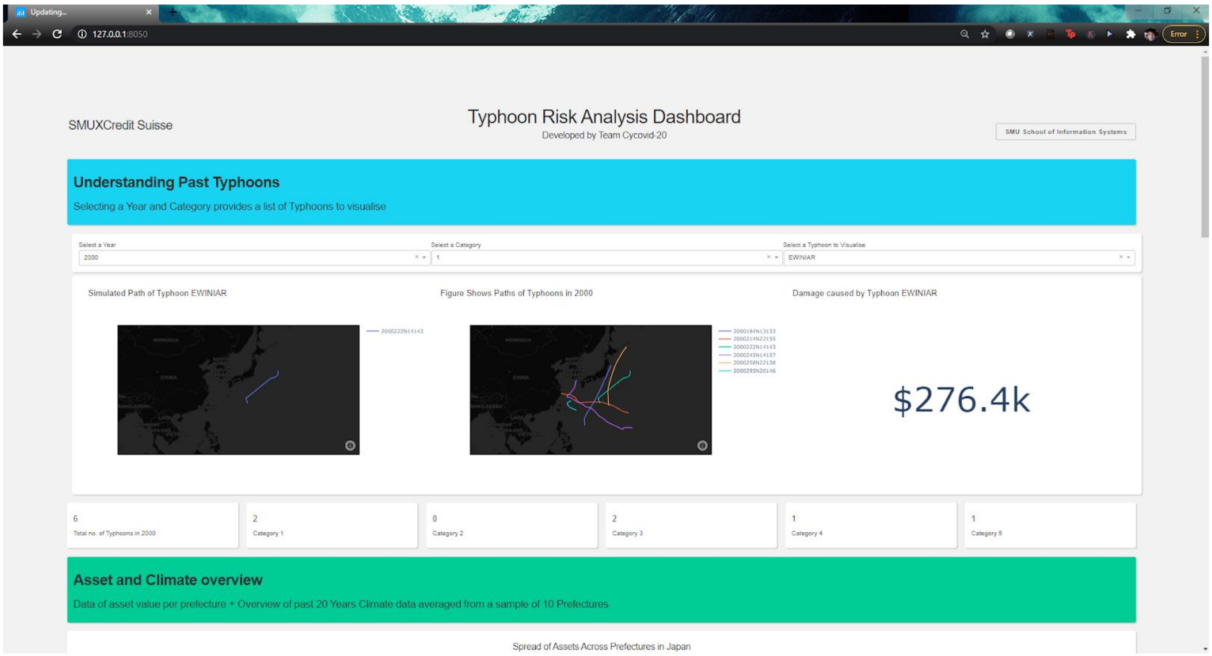
This will activate the command line. In the virtual environment’s command line, the user will enter the local directory where he or she has installed the application.
E.g. C:\Users\Cycovid_Student\Desktop\Application_Files (Change depending on where the folder is stored)



Once the user is here, type the command:
Python Typhoon_Simulator.py
In a few moments, this message will appear.



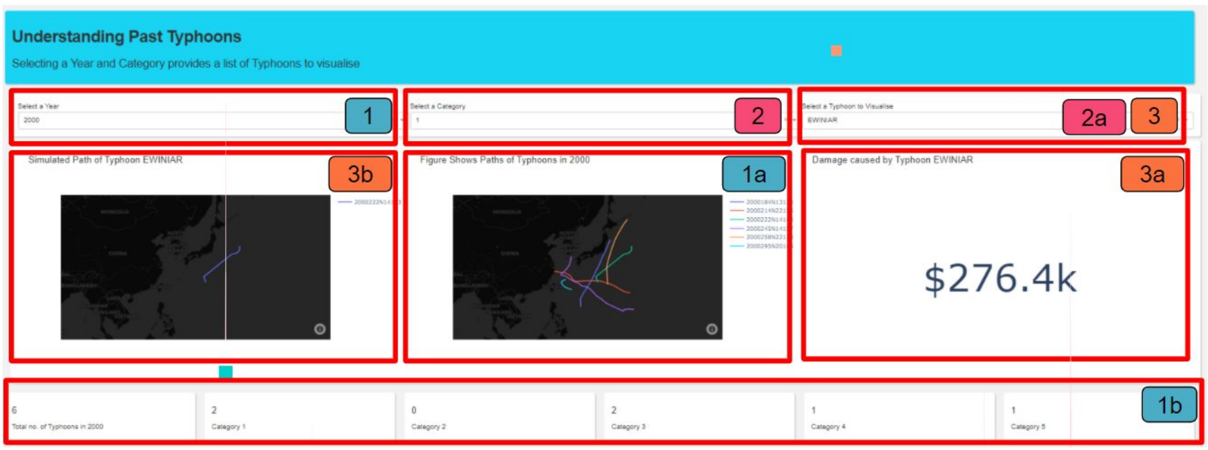
Once the command line says “running on http://127.0.0.1:8050/” the application can be accessed from the web browser. Simply open any browser and type in http://127.0.0.1:8050/ as the url and the application should be shown up as shown below.



Once the user reaches this step, the application is successfully installed.

In the next portion, we will look at the various functionalities of our application individually.

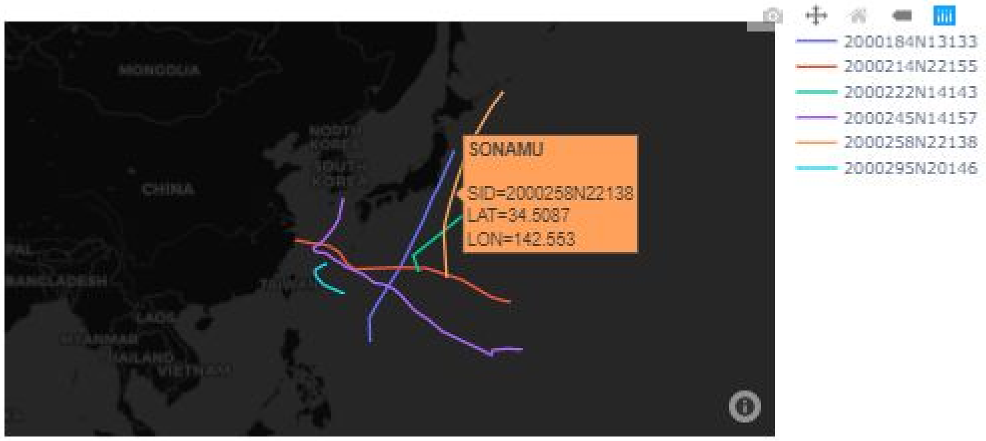
6.2.2 Analysis of Historical Typhoons



[1] The user first selects the year which they would like to view. This changes the paths visualised in [1a] as well as the count of typhoons displayed on the cards in [1b]. Next, the user is able to further filter down the typhoons based on category [2] naturally. The default value of the list is category 1. However, upon user’s selection of category (1 to 5), the drop-down list in [2a] will be altered such that it only shows typhoons that belong to the year selected in [1] that are under the category selected in [2]. In the event that the list is empty, the dropdown box will display “no results found” and the map in [3b] will be blank. Upon

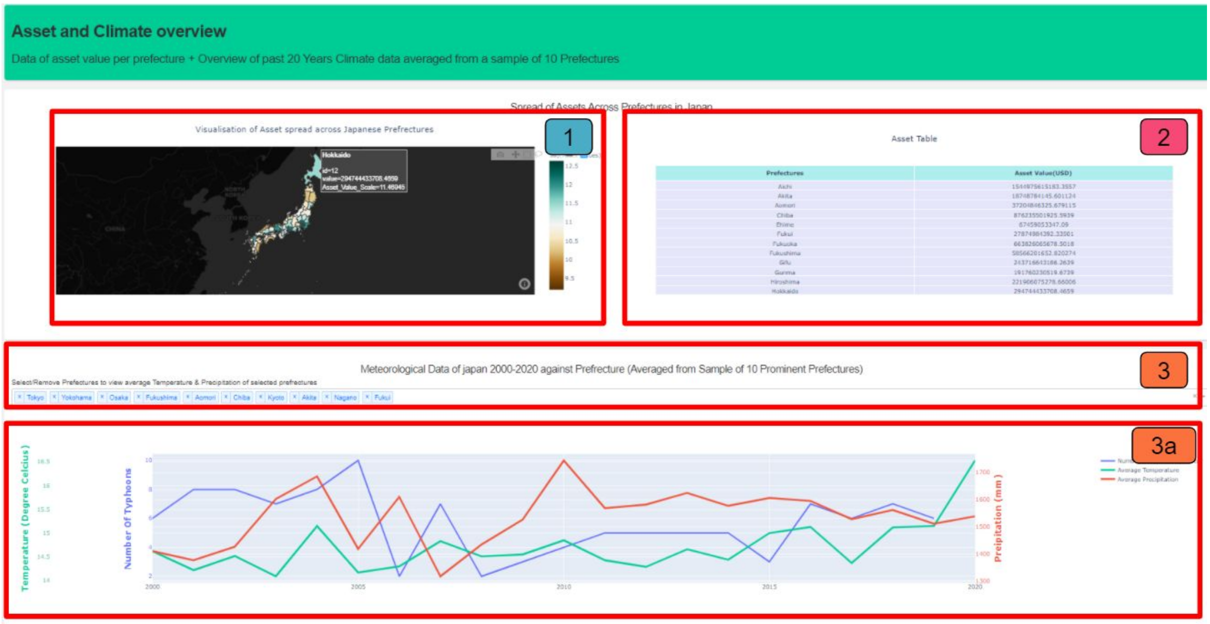
selection of a typhoon in [3], the specific path of the typhoon will be visualised in [3b] and the calculated damages from the team's research will be shown in [3a].

Figure Shows Paths of Typhoons in 2000



Additionally, when hovering over the map in [1a], the user will be able to see the names of the typhoons that occurred in the selected year [1].

6.2.3 Asset and Climate Overview

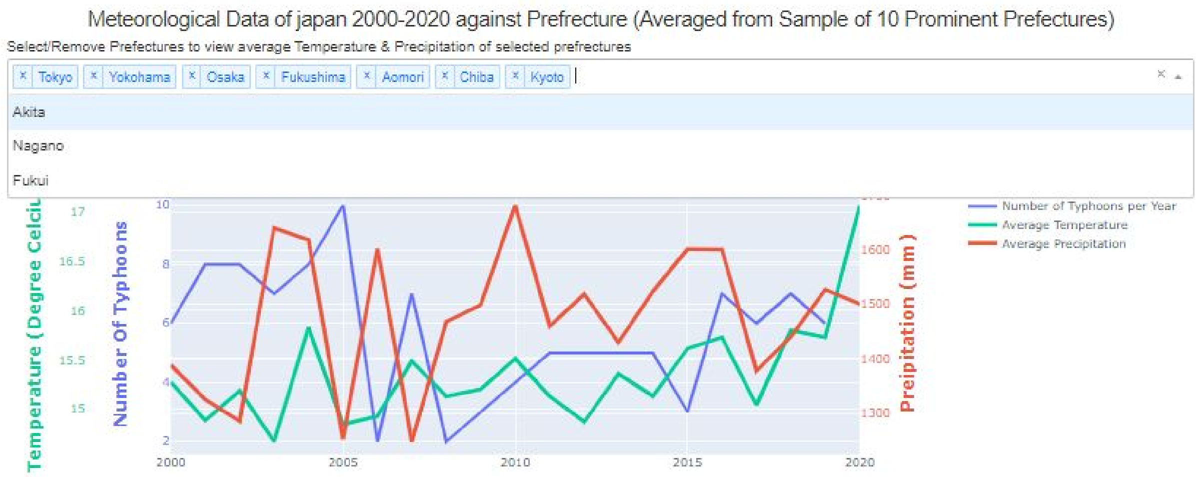


While [1] & [2] allows the user to better understand asset exposure in Japan, [3] focuses on climate changes and the correlation to the number of typhoons.

[1] The user is able to observe the spread of assets across Japan. Since the asset disparity was very large, the Choropleth Map's Colouration was done using log(actual asset value) to provide a better representation for the different regions. By hovering over each prefecture, the actual asset value is shown as well.

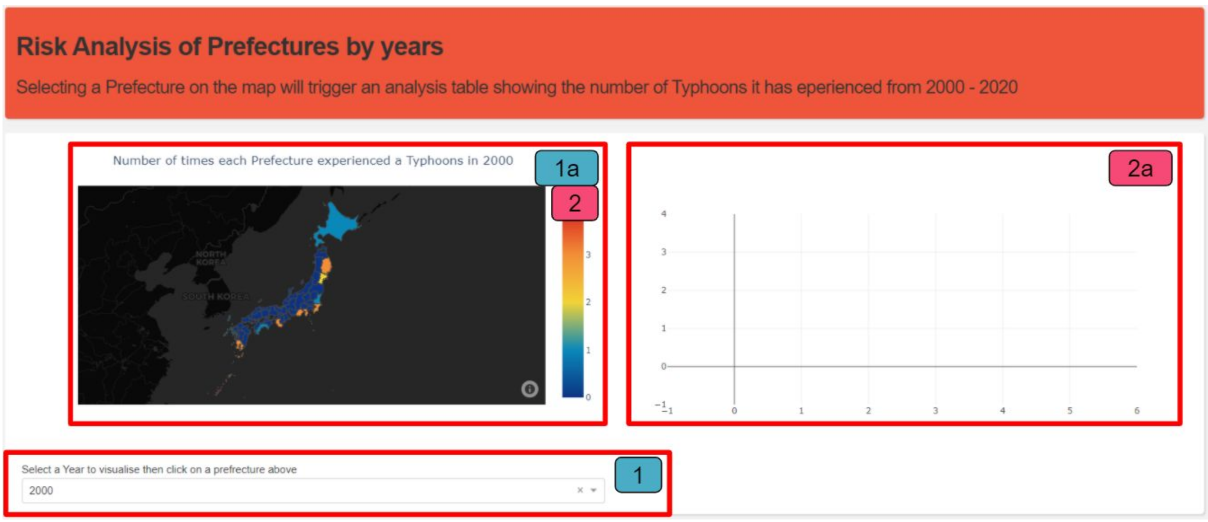
[2] In case the user is checking on a specific prefecture, the map in [1] would be less useful. Hence, an alphabetically ordered table of the prefectures and the asset value is shown.

[3] The default setting of the application selects all available prefectures in the dropdown box. The annual precipitation and temperatures of these prefectures are then averaged and visualised in the line graph in [3a] (green - temperature, red - precipitation). The user is able to remove prefectures by clicking on the cross in [3]. This will remove the prefecture's value from the line graph and a new average is then generated in [3a]. The blue line in [3a] represents the number of typhoons that occurred in a given year. Since it is a historical record, the value is stagnant.



Additionally, the unselected prefectures can be reselected by clicking on the dropdown box in [3].

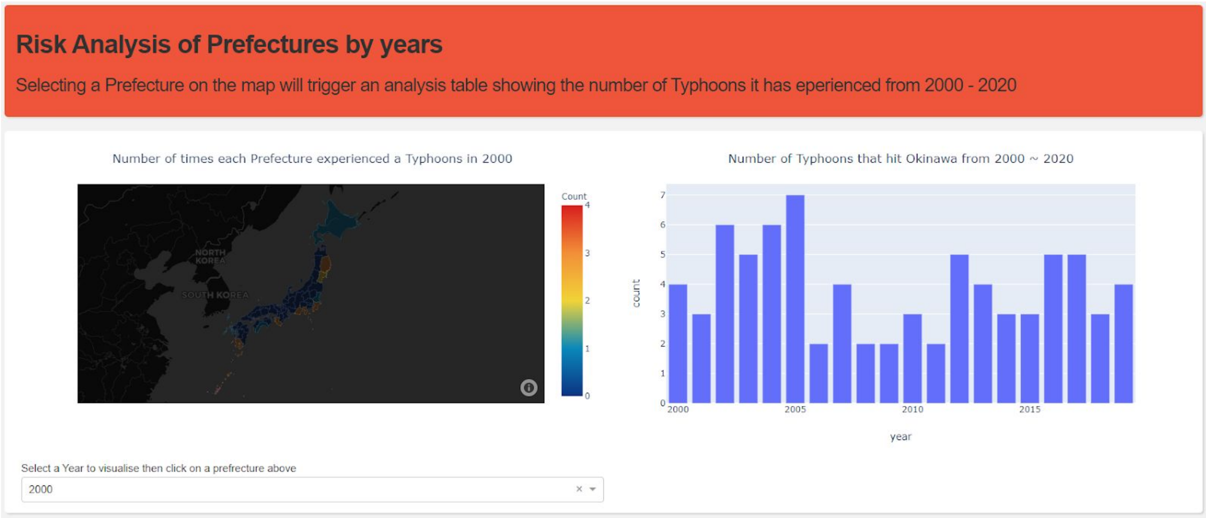
6.2.4 Risk Analysis Of Prefectures Across Japan



[1] The user will select a year to visualise (the default value is always 2000). This will alter the visualisation in [1a]. [1a] Shows a choropleth map of Japanese Prefectures with respect

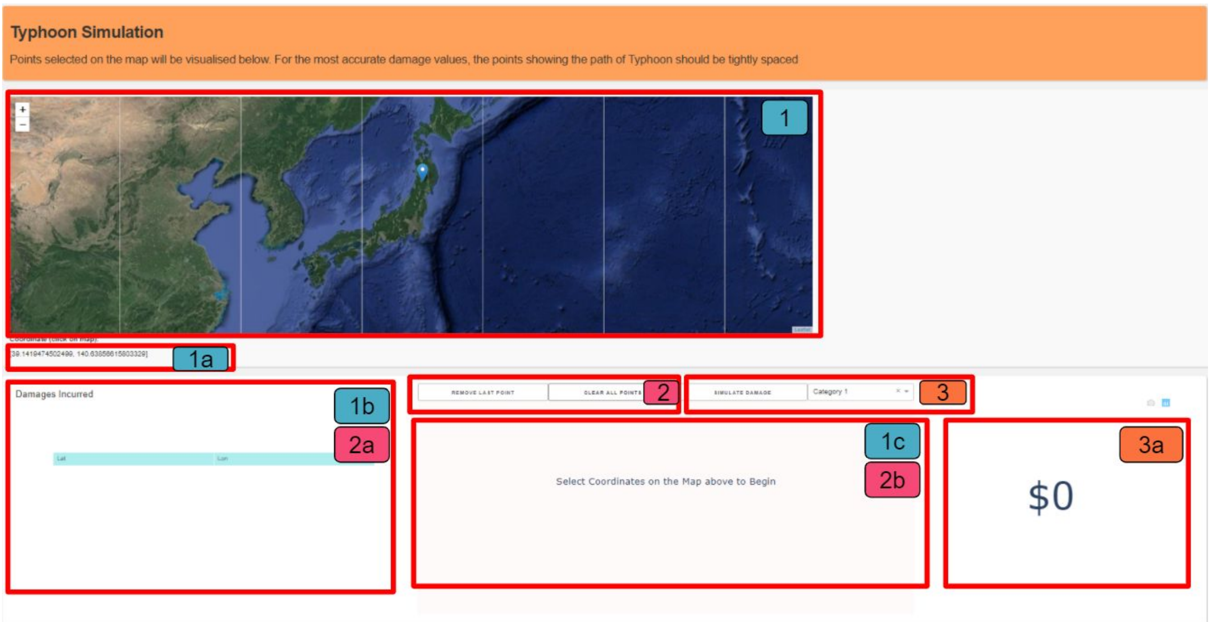
to the number of times a specific prefecture has been hit by a typhoon in the selected year [1].

[2] While hovering over the map, the user will have the functionality to select a prefecture to inspect. This triggers a new table to show in [2a]. Here is what it looks like.

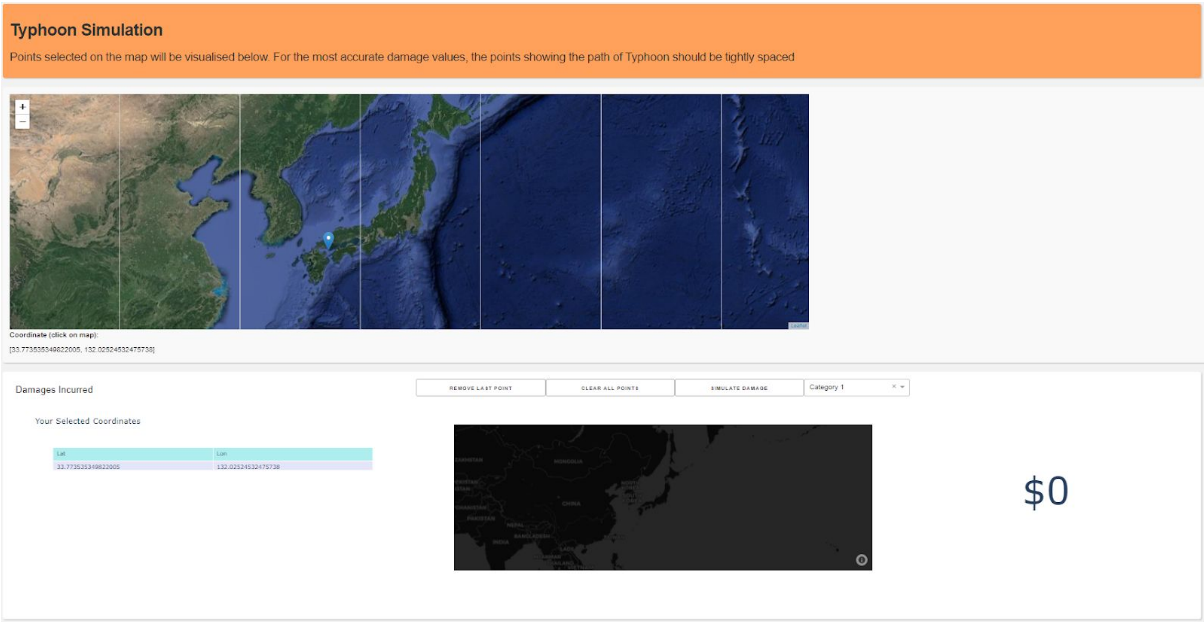


The table shows the annual number of times a prefecture gets hit by a typhoon from the year 2000 until the present time. This table allows the user to assess the frequency of typhoons with respect to other prefectures so as to make better-informed business decisions.

6.2.5 Typhoon Simulator



[1] The user clicks on coordinates on the map. Upon selection, the latest coordinate is shown on [1a] to validate that the user’s clicks are being recorded. The data is also added to the table in [1b] as shown below.

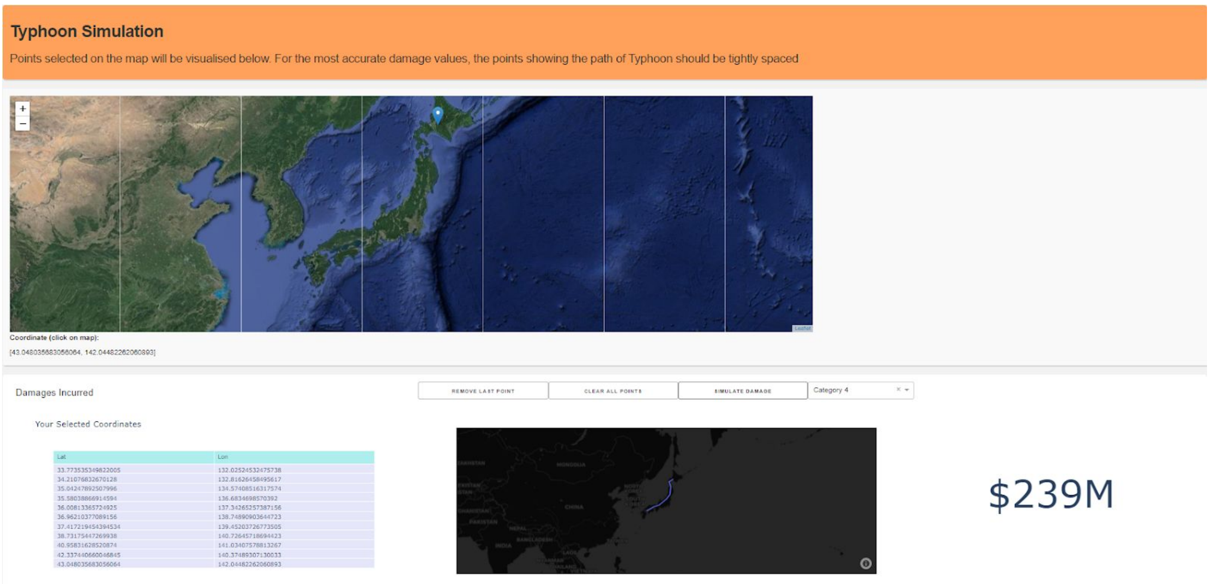


Upon selection of a coordinate in [1], a map is shown in [1c]. This map is used to simulate the user's synthetic typhoon based on click input. However, the path will only be shown when more than 1 coordinate is in the table [1b].

[2] This portion allows the user to manipulate the selected coordinates. The user is able to either "remove last point" or "clear all points" previously selected. Upon clicking, either the last entry will be removed from table in [2a] or the entire table will be emptied. Similarly, the path of a typhoon will be updated in [2b] with either the last point being removed or the entire path being cleared.

[3] Once a path has been selected by the user in [1], the user can select a Typhoon category to simulate. The dropdown box shows the options for category (1 to 5) and after selecting the category, the user clicks on simulate damage. This will generate a damage value in [3a].

An example of the simulated path and damage is shown below.



To further elaborate on the simulation:
The selected coordinates are reverse geocoded using Big Data Cloud API and cleaned instantaneously using backend python logic. The damages are then simulated through the team's algorithm to determine a monetary value. Big Data Cloud API was used as opposed to the reverse geocoder python library due to the speed of data retrieval. When the reverse geocoder library was used, the application was too slow to function normally. Hence, we wrote python functions to streamline the json data from Big Data Cloud allowing us to tap on its speed while maintaining accurate queries.

6.3 Evaluation Models

In order to check the accuracy of the application, the team used an F-Test to check if there is an even distribution of variance between the simulated data and the actual data.

Due to a lack of transparency for the monetary impact of typhoons in Japan, we were limited to sampling those that were represented. The team placed the data for actual and simulated in a side by side comparison and got the following results.

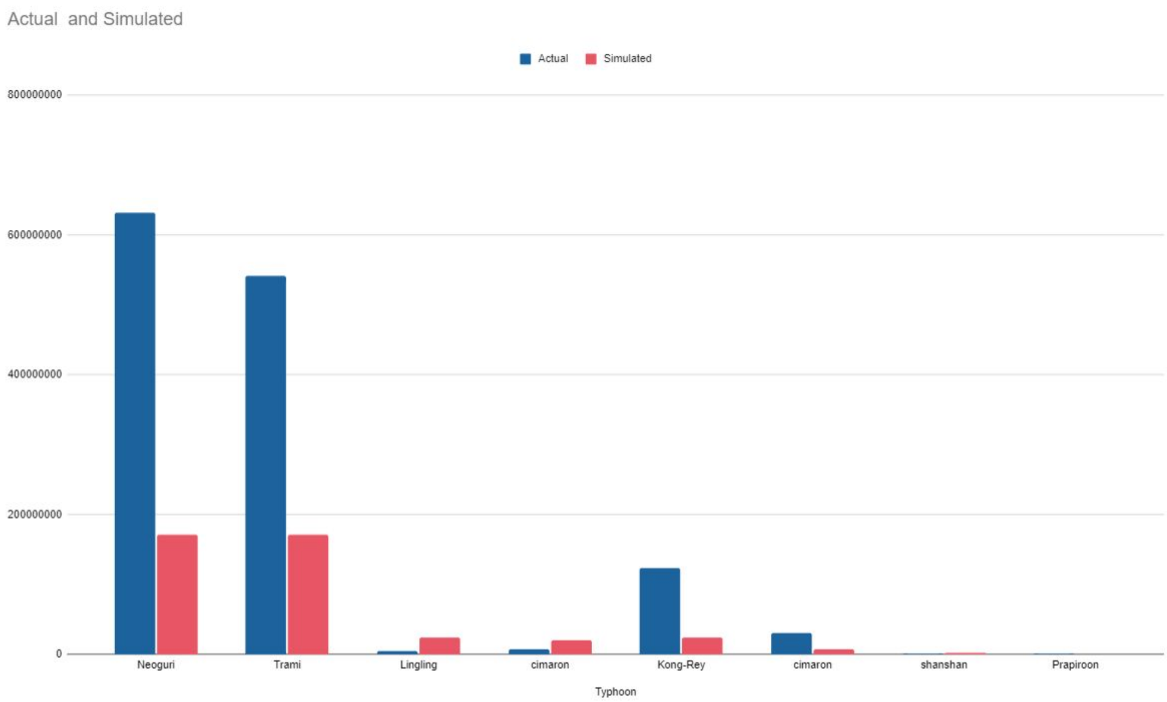


Figure 46: Damage data visualised in bar chart

Typhoon	Actual	Simulated	Cat	Year
Neoguri	632000000	171200000	5	2014
Trami	542000000	171300000	5	2018
Lingling	4980000	23610000	4	2019
cimaron	6680000	20600000	4	2017
Kong-Rey	123000000	23610000	4	2018
cimaron	30600000	6680000	4	2018
shanshan	866000	1842000	2	2018
Prapiroon	446000	93100	1	2018

Figure 47: Damage data visualised in a table

Due to the nature of the team's study, the asset value only considered produced capital. However, upon documentation done by well-equipped institutions, the recorded damages consisted of loss of human lives, future impact on the industry, economic downtime and more. Hence, it is reflected by unproportionate values. To better evaluate our data, the team decided to normalise the values using log10 to place them on the same scale as seen below *(Figure 48)*. The team then ran the values through a Two-Sample for Variances F-Test.

Typhoon	log(actual)	log(simulate)	Actual	Simulated	Cat	Year
Neoguri	8.8007171	8.23350376	632000000	171200000	5	2014
Trami	8.7339993	8.23375736	542000000	171300000	5	2018
Lingling	6.6972293	7.37309599	4980000	23610000	4	2019
cimaron	6.8247765	7.31386722	6680000	20600000	4	2017
Kong-Rey	8.0899051	7.37309599	123000000	23610000	4	2018
cimaron	7.4857214	6.82477646	30600000	6680000	4	2018
shanshan	5.9375179	6.26528963	866000	1842000	2	2018
Prapiroon	5.6493349	4.96894968	446000	93100	1	2018

F-Test Two-Sample for Variances

	Variable 1	Variable 2
Mean	7.277400182	7.073292011
Variance	1.447583571	1.153328127
Observations	8	8
df	7	7
F	1.25513593	
P(F<=f) one-t	0.385983238	
F Critical one-	3.78704354	

Figure 48: Normalised data and Two-Sample for Variances F-Test

Looking at the results, since F < F Critical one-tail, we accept the null hypothesis. This means that the variances of the two populations are equal.

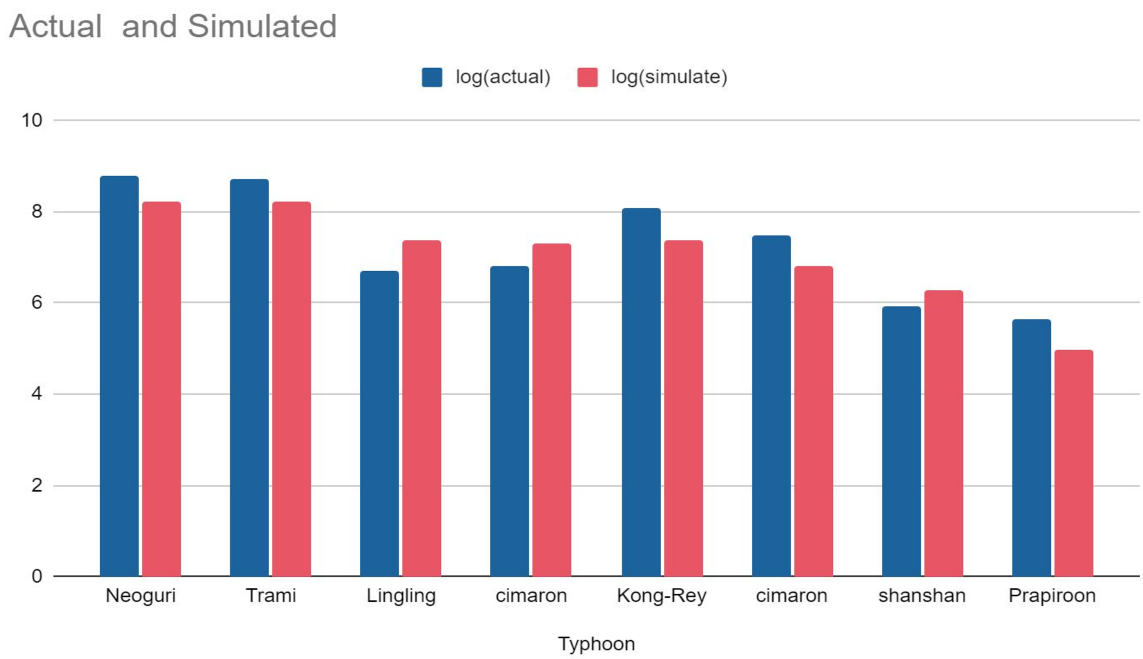


Figure 49: Normalised data actual vs simulated comparison

Furthermore, a comparison of the log10(actual) vs log10(simulated) shows similar trends of rises and falls in data (*Figure 49*). Hence, we have substantial proof to validate the accuracy of the results.

6.4 Quality Attributes (KPI)

6.4.1 Usability

The final application has managed to garner positive feedback from the users during the second round of User Acceptance Testing (UAT) with an average of 6.5 minutes taken for each user to complete the assigned tasks. The two groups enrolled in the first and second round of UAT are completely independent of one another and the application appears to be relatively easy to understand for first-time users (after the first round of UAT), given that there was a significant decrease of 3.5 minutes in the average completion time. With a minimalistic user interface with clearly labelled tables and graphs, the application is rather intuitive and easy to navigate.

6.4.2 Maintainability

The application has been carefully segmented into different folders for ease of management. The codes are divided into their respective pages and labelled with comments such that it would be easy for users to navigate around. The users would be able to find corresponding codes with a simple lookup should they want to alter anything. The team has included a YML file for the easy creation of virtual environments and automated installations of required libraries. The file also enables users to get a good overview of the libraries and their respective versions such that the need for version updates can be identified. Hence, future users of the application would be able to render a fully functioning application without the need of paying additional license fees.

6.4.3 Scalability

The current application centers around the typhoons in Japan due to the time and data limitation of the project. However, it is easily scalable to a global magnitude if the application were to be subjected to a larger dataset. The damage function generated can be altered with the utilisation of the GDP values obtained from various countries, such that corresponding impacts can be generated.

7. Project Management

7.1 Timeline

Based on the scheduled timeline, the team planned to finish the application by Week 13 and focus on the final deliverables such as the poster and report for the rest of the weeks *(Figure 50).*

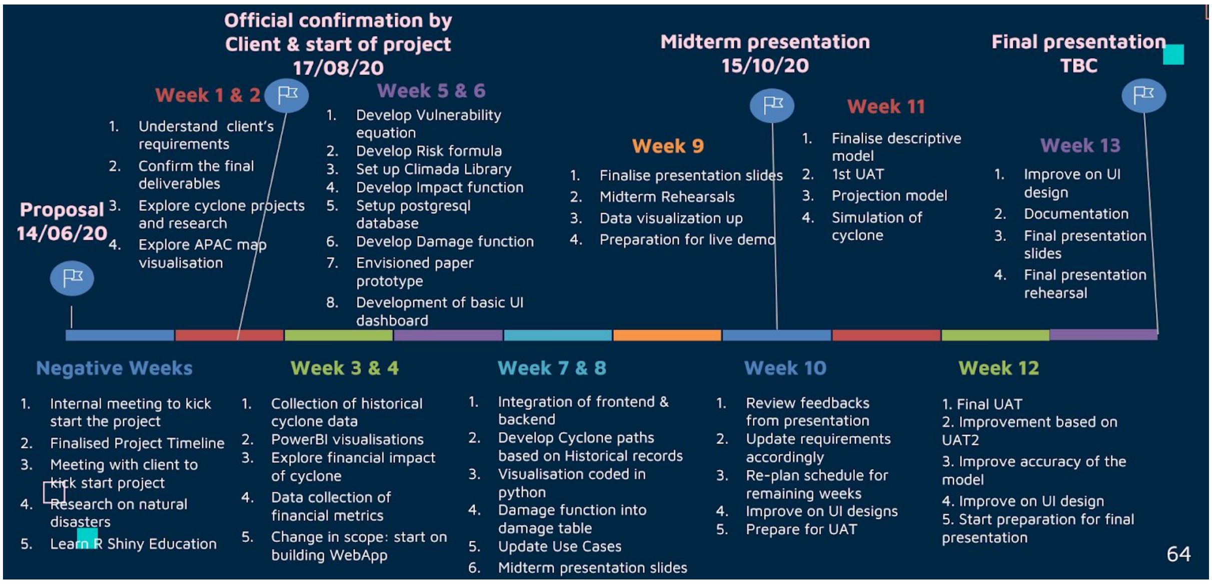


Figure 50: Scheduled timeline

However, Week 11 to 13 was spent on additional research studies while building the application. Below is the revised timeline. The team started the preparation for UAT and final presentation in week 13 *(Figure 51)*.

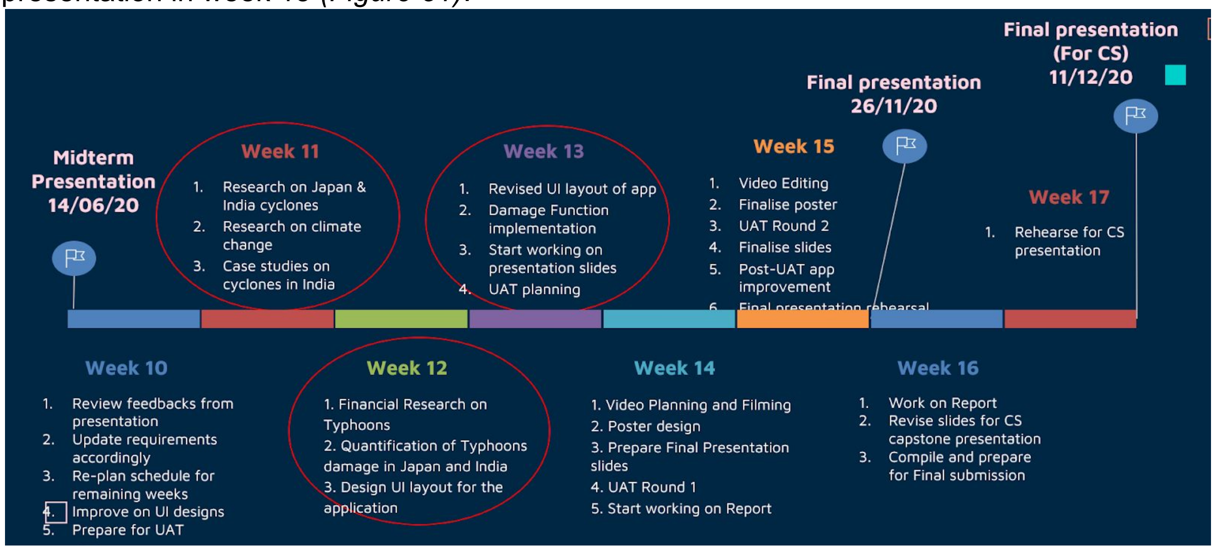


Figure 51: Revised timeline

7.2 Approach

Before Midterms, the team was having weekly scrum to update each other about the progress. But after the midterm presentation, we decided to switch to 2 day sprints which increased our efficiency significantly. With the team working closely together, we were more focused on the tasks and able to solve problems as a team. This was also the approach that ensures everyone is on the same page.

7.3 Scope Change

In terms of scope, the team shifted the focus from the Philippines to India and Japan with the reasons mentioned in the earlier section. The team also took feedback from our client seriously by focusing on the research study suggested. After looking into the impact of climate change on typhoons and having a discussion with our client, we decided to include climate change in both our research study and the web application.

For architecture, initially, the team wanted to create an application with front-end and back-end connected to a database and server, hence we explored and played around with several different frameworks. However, due to the multiple shifts in the overall scope and focus, we have changed to creating a dashboard UI using plotly and dash. For the database, instead of storing records, we have opted for the dashboard to make use of uploaded csv files for easy access and manipulation.

7.4 Stakeholder Management

The point of contact from Credit Suisse is Mr. Huang Sile. The team met with him online about once a week to give him our updates and discussed the area of improvements and direction of the project. The team also met with our Supervisor Professor Wang Hai weekly to show him the timeline on the progress and what we were working on for the project. Lastly, within the team, we met up very frequently after the midterm presentation since we switched to a 2 day sprint approach as well as the increased workload from the scope changes.

The team opted for the use of google drive to manage all the documents such as meeting minutes in an orderly manner. Besides that, to better manage and facilitate our team's progress, we have been using Clubhouse, which is a project management platform for us to plan the sprint iterations, task allocations, bug metrics, etc.

8. Gap Analysis & Future Work

8.1 Gap Analysis

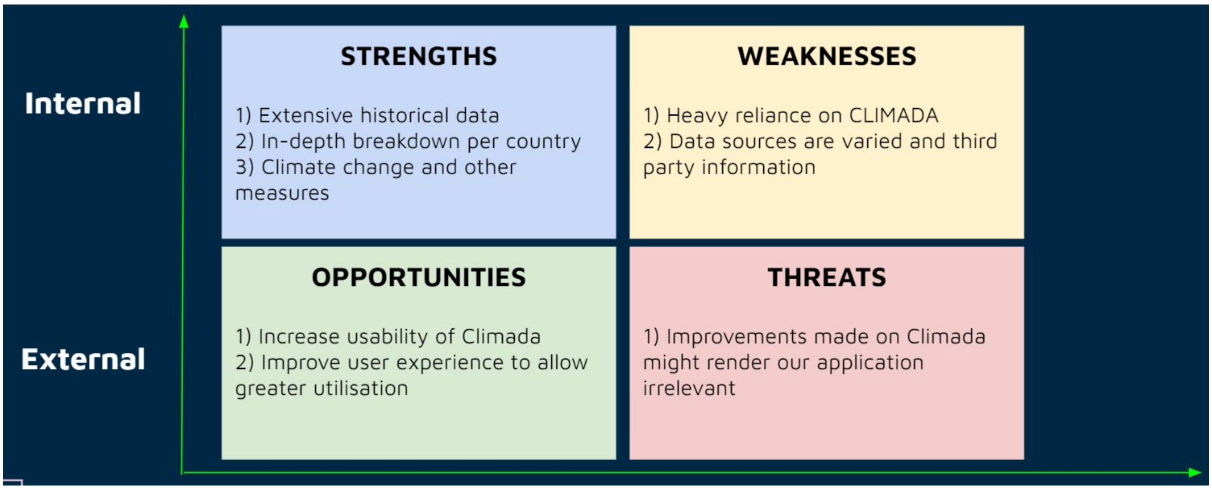


Figure 52: Gap analysis (SWOT)

The team understands that the proposed solution is not perfect and hence, we performed a gap analysis on our solution *(Figure 52)*.

Strengths: The team has performed extensive research and analysed a vast amount of historical data on typhoons in Japan and cyclones in India as covered in the Research portion above. In addition, we did an in-depth breakdown based on the two countries as well as climate change factors that caused an impact to typhoons/cyclones.

Weaknesses: The team does rely heavily on the exposure class of Climada because there was inconsistent data or documentation of Produced Capita on the net. Therefore, we have to use the algorithm on Climada to estimate the Produced Capita and damaged assets on Climada. On top of that, the team also has multiple data sources and the research took up a long period of time because most of the typhoons/cyclones data were not being standardised and we aim to ensure that we retrieve the relevant information needed from credible sources.

Opportunities: The team can increase the use of Climada as there are still many functions and algorithms that we have yet to use in our application such as other assets like foreign reserves and human capital. The team can also further improve the user experience and usability of the application by conducting more rounds of UAT.

Threats: The threat the team faced was mainly on Climada. As mentioned before, the application relies heavily on the use of Climada. Any changes to their fundamental features for instance the way they arrange their data sets etc will result in many changes to be made in order for us to keep the application relevant in the next 5-10 years.

8.2 Future Work

Referring back to the Gap Analysis above, the team has identified the opportunities and growth for the application. If there is the next phase of this project, we can create dashboards for other countries and take into consideration other assets like the Foreign Reserves and Human Capital to calculate the total asset damage. Additionally, the team can also incorporate a data scraper for the application in order to resolve and automate the process of having to do extensive research over multiple websites for the total damage caused by typhoons/cyclones in different countries. Lastly, we can explore methods to make the application model more intricate and improve the overall accuracy of our predictions.

9. Conclusion

9.1 X-Factor

- Plotting of typhoon paths:
  The team was able to clean and analyse the IBTrACS data and plot the relevant typhoon paths that the user selects.
- Immediate generation of typhoon damages:
  The application synchronously calls various APIs and uses the damage function we came up with to generate the asset damage instantly on the asset table visualisation.
- Highly scalable:
  The application is highly scalable as we can easily apply the analysis the team has performed for Japan on any other countries of interest. It can also be used by the risk assessment team in any Credit Suisse office around the world.

9.2 Benefits for Sponsor

Credit Suisse would be able to analyse the spread of assets across the country. For the project, the team has focussed on Produced Capital. By understanding the prefectures that are most impacted, they can actually locate which are the companies that have their main operations there. Additionally, they will also be able to understand the market sentiments in the different prefectures that were affected by typhoons. Therefore, they will be able to assess their lending position with the company based on that. Lastly, managing the risk of future investments using the climate change data that we have done will enable them to understand the correlation between climate change and the impacts of typhoons. Using the research study on India and the meteorological data given on the application, Credit Suisse will be able to see the trend of climate change and make smart investments for the next decade.

9.3 Team Effort

- Everyone in the team is willing to share their expertise and contribute their effort so that everyone can benefit and learn together as a team.
- Each of us has different strengths and weaknesses where we try to complement one another to get the best efficiency.
- The team is willing to compromise on our personal schedule and meetup on weekends to complete the project deliverables.
- The team tried our best to help one another when any problem arises and leave no one behind.
- Everyone understands their roles and responsibilities and makes an effort to complete the tasks required.

9.4 Learning Takeaways

- Financial Risk varies across geographies and research should be done properly before any investment takes place.
- Learn and adapt to work with Scrum, which is an Agile Methodology for us to stay focused in order to deliver the project requirements in the shortest time possible.
- Climate change varies from different countries and there is a correlation in relation to investments as well.
- The team also learned and explored things outside of what was being taught in school (the use of different software, tools, and libraries).
- The team had an opportunity to work on and build a full-scale application from scratch.
- The team got to interact with real world clients and adapt to their application requirement needs.
- The team learned to deal and adapt with constant scope changes in a real-world environment.

10. References

https://www.credit-suisse.com/about-us/en/our-company/who-we-are.html

https://www.investopedia.com/terms/g/gdp.asp

https://climada-python.readthedocs.io/en/stable/tutorial/climada_entity_LitPop.html

https://indiaclimatedialogue.net/2020/06/05/cyclones-rise-as-climate-change-heats-up-indian-ocean/

https://reliefweb.int/map/world/asia-pacific-regional-hazard-map-tropical-storm-risk

https://cdn-images.kontinentalist.com/static-html/asia-pacific-natural-disaster-risk-reduction-and-management/index.html

http://www.rsmcnewdelhi.imd.gov.in/images/pdf/publications/preliminary-report/amphan.pdf

https://www.carbonbrief.org/global-warming-has-changed-spread-of-tropical-cyclones-around-the-world

https://en.wikipedia.org/wiki/Tropical_cyclones_and_climate_change

https://www.noaa.gov/news/study-climate-change-has-been-influencing-where-tropical-cyclones-rage

https://www.gfdl.noaa.gov/global-warming-and-hurricanes/

https://yaleclimateconnections.org/2019/07/how-climate-change-is-making-hurricanes-more-dangerous/

https://www.nature.com/articles/s41598-020-58824-8

https://www.sciencedirect.com/science/article/pii/S2225603220300047

https://www.sciencedirect.com/science/article/pii/S222560321830033X

https://www.researchgate.net/publication/42757757_Tropical_Cyclones_and_Climate_Change

https://www.newindianexpress.com/nation/2020/jun/03/climate-change-making-cyclones-fiercer-more-frequent-researchers-2151411.html

https://www.sciencedaily.com/releases/2020/05/200504155207.htm

https://ohiostate.pressbooks.pub/sciencebites/chapter/climate-change-opens-the-door-to-more-intense-tropical-storms/

https://www.pnas.org/content/117/20/10706

https://ocean-climate.org/?p=4832&lang=en

https://www.jstor.org/stable/24868735?seq=1

https://timesofindia.indiatimes.com/india/32-rise-in-number-of-cyclones-in-past-5-years-imd/articleshow/72031005.cms

https://www.indiatvnews.com/fyi/news-all-you-need-to-know-about-cyclones-in-india-most-affected-states-list-of-9-deadliest-cyclones-in-india-519361

http://www.rsmcnewdelhi.imd.gov.in/images/pdf/climatalogy/lifeperiod.pdf

https://www.indiatvnews.com/fyi/news-all-you-need-to-know-about-cyclones-in-india-most-affected-states-list-of-9-deadliest-cyclones-in-india-519361

https://timesofindia.indiatimes.com/india/32-rise-in-number-of-cyclones-in-past-5-years-imd/articleshow/72031005.cms

https://climate.nasa.gov/vital-signs/global-temperature/

https://www.data.jma.go.jp/obd/stats/etrn/view/monthly_s3_en.php?block_no=47412&view=13