Developing a Smart Traveler Insights System to Analyse Post Pandemic Tourism Recovery in Asia Pacific

By

ABDUL MUHAIMIN AMAN

TP069510

APD3F2502CS(DA)

A report submitted in partial fulfillment of the requirements for the degree of

B.Sc. (Hons) Computer Science Specialism in Data Analytics

at Asia Pacific University of Technology and Innovation.

Supervised by Ms. Fatin Izzati Ramli

2nd Marker: Ms. Mary Ting

2025

DECLARATION OF THESIS CONFIDENTIALITY

Author's full name: **ABDUL MUHAIMIN AMAN**

IC No./Passport No.: **W0637982**

Thesis/Project title: **DEVELOPING A SMART TRAVELLER INSIGHTS SYSTEM TO ANALYSE POST PANDEMIC RECOVERY IN ASIA PACIFIC**

I declare that this thesis is classified as:

☐ CONFIDENTIAL

☑ RESTRICTED

☐ OPEN ACCESS

I acknowledged that Asia Pacific University of Technology & Innovation (APU) reserves the right as follows:

1. The thesis is the property of Asia Pacific University of Technology & Innovation (APU).
2. The Library of Asia Pacific University of Technology & Innovation (APU) has the right to make copies for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Author's Signature: *AMAN*

Date: 17 September 2025

Supervisor's Name: Ms. **FATIN BIN IZZATI**

Date: 17 September 2025

Signature: .....

Please fill in **all** the following details for library cataloguing purposes.

First Name: **Abdul Muhaimin**
Middle Name (only if applicable) :
Last Name: Aman
Title of the Final Year Project / Dissertation / Thesis : **DEVELOPING A SMART TRAVELLER INSIGHTS SYSTEM TO ANALYSE POST PANDEMIC RECOVERY IN ASIA PACIFIC**
ABSTRACT Unprecedented disruptions in international travel brought about by the COVID-19 pandemic presented methodological and practical difficulties for recovery forecasting. Under structural shocks, conventional econometric and univariate models—which presuppose stable seasonal patterns—proved insufficient. In order to predict the post-pandemic tourism recovery in Asia-Pacific cities, this study creates the Smart Traveller Insights system, a data-driven decision-support tool. Because of their worldwide importance and the availability of reliable monthly data, three locations—Hong Kong, Singapore, and Bangkok—were chosen as case studies. Through the creation of a web-based platform called the Smart Traveller Insights System, which uses real-time data to evaluate traveller confidence in a few Asia-Pacific cities, this project seeks to close this gap. The platform offers useful insights grouped by tourism types, including luxury, cultural, and eco-tourism, by examining search trends, social media sentiment, and tourism data. According to preliminary research, these data-driven tools can help people make safer and better-informed travel choices while also identifying cities that require assistance during the recovery process. Benchmarked against a Seasonal-Naïve baseline, the methodology combines machine learning and time-series techniques such as Ridge regression, Random Forest, and XGBoost. Lagged arrivals, rolling means, seasonal dummies, and a COVID-period dummy were all included in feature engineering. MAE, RMSE, MAPE, and R2 were used to assess the models' performance after they were validated on the 2024 hold-out year and trained on pre-pandemic and partial recovery data (2017–2023). The findings showed that Seasonal-

Naïve was most defendable for Hong Kong (MAPE = 10.10%), Random Forest was most dependable for Singapore despite low variance in test data (MAPE = 5.04%), and Ridge regression performed well in Bangkok (R2 = 0.859, MAPE = 3.09%). Despite its theoretical strength, XGBoost showed weaker generalization and overfitting.

The study operationalized its findings into a repeatable deployment pipeline in addition to forecasting. In order to create auditable artifacts, the backend modularized data ingestion, feature engineering, evaluation, and scenario generation. These artifacts were displayed by the Streamlit-developed frontend as an interactive dashboard featuring model cards, tabular outputs, time-series visualisations, headline metrics, and download options. Transparency, interpretability, and conformity to current best practices in responsible AI and forecasting were all guaranteed by this design.

The Smart Traveller Insights system makes a methodological and practical contribution. It proves that in times of crisis, accurate short-term tourism forecasts can be produced through rigorous validation and context-sensitive model selection. It also demonstrates how scholarly research can be operationalized into a useful tool for decision-making, which could help industry and policymakers plan for a sustainable tourism recovery.

By encouraging safer, more assured travel in the post-pandemic era, this work supports Sustainable Development Goal 3: Good Health and Well-Being.

Keywords: Tourism forecasting, post-pandemic recovery, machine learning, Ridge regression, Random Forest, Seasonal-Naïve, Streamlit dashboard, Asia-Pacific.

General Subject: Computer Science Data Analytics
Date of Submission : 17th September 2025

ACKNOWLEDGEMENT

Embarking on the journey of accomplishing the goals of this project and completing this (IR) has been a profoundly enlightening and life-changing experience. In the months spent conducting research, analysis, and development, I have developed my technical expertise as well as my ability to think critically, persevere, and comprehend real-world problems in the post-pandemic tourism industry. I was forced to apply data-driven methodologies with academic rigor and go beyond my theoretical knowledge for this project, which helped me become a more capable and perceptive researcher.

I want to sincerely thank my supervisor Ms. Fatin Izzati Ramli and my 2nd marker Ms. Mary Ting whose constant encouragement, helpful criticism, and unwavering support were invaluable in helping me overcome many obstacles and fine-tune the project's course. Their advice greatly improved the calibre of this report.

My family's unwavering patience, support, and faith in me gave me the emotional fortitude I needed to finish this difficult journey, and for that I am incredibly grateful. I also want to thank my friends and peers for their insightful comments, helpful edits, and moral support during this process.

I have learned so much from this project about discipline, resilience, and intellectual curiosity in addition to data science, tourism recovery, and system development. I sincerely appreciate all of the advice, criticism, and encouragement that helped me reach this significant academic milestone.

ABSTRACT

Unprecedented disruptions in international travel brought about by the COVID-19 pandemic presented methodological and practical difficulties for recovery forecasting. Under structural shocks, conventional econometric and univariate models—which presuppose stable seasonal patterns—proved insufficient. In order to predict the post-pandemic tourism recovery in Asia-Pacific cities, this study creates the Smart Traveller Insights system, a data-driven decision-support tool. Because of their worldwide importance and the availability of reliable monthly data, three locations—Hong Kong, Singapore, and Bangkok—were chosen as case studies.

Through the creation of a web-based platform called the Smart Traveller Insights System, which uses real-time data to evaluate traveller confidence in a few Asia-Pacific cities, this project seeks to close this gap. The platform offers useful insights grouped by tourism types, including luxury, cultural, and eco-tourism, by examining search trends, social media sentiment, and tourism data. According to preliminary research, these data-driven tools can help people make safer and better-informed travel choices while also identifying cities that require assistance during the recovery process.

Benchmarked against a Seasonal-Naïve baseline, the methodology combines machine learning and time-series techniques such as Ridge regression, Random Forest, and XGBoost. Lagged arrivals, rolling means, seasonal dummies, and a COVID-period dummy were all included in feature engineering. MAE, RMSE, MAPE, and R2 were used to assess the models' performance after they were validated on the 2024 hold-out year and trained on pre-pandemic and partial recovery data (2017–2023). The findings showed that Seasonal-Naïve was most defendable for Hong Kong (MAPE = 10.10%), Random Forest was most dependable for Singapore despite low variance in test data (MAPE = 5.04%), and Ridge regression performed well in Bangkok (R2 = 0.859, MAPE = 3.09%). Despite its theoretical strength, XGBoost showed weaker generalization and overfitting.

The study operationalized its findings into a repeatable deployment pipeline in addition to forecasting. In order to create auditable artifacts, the backend modularized data ingestion, feature engineering, evaluation, and scenario generation. These artifacts were displayed by the Streamlit-developed frontend as an interactive dashboard featuring model cards, tabular outputs, time-series visualisations, headline metrics, and download options. Transparency,

interpretability, and conformity to current best practices in responsible AI and forecasting were all guaranteed by this design.

The Smart Traveller Insights system makes a methodological and practical contribution. It proves that in times of crisis, accurate short-term tourism forecasts can be produced through rigorous validation and context-sensitive model selection. It also demonstrates how scholarly research can be operationalized into a useful tool for decision-making, which could help industry and policymakers plan for a sustainable tourism recovery.

By encouraging safer, more assured travel in the post-pandemic era, this work supports Sustainable Development Goal 3: Good Health and Well-Being.

Keywords: Tourism forecasting, post-pandemic recovery, machine learning, Ridge regression, Random Forest, Seasonal-Naïve, Streamlit dashboard, Asia-Pacific.

Table of Contents

Table of Figures

List of Tables

CHAPTER 1: INTRODUCTION

1.1 INTRODUCTION

The COVID-19 pandemic caused an unprecedented collapse in international tourism, with the steepest decline on record occurring in 2020, when global international arrivals fell by roughly 74% compared to 2019 (UNWTO, 2021). The need for forward-looking forecasting tools rather than lag descriptive indicators is further supported by the Asia-Pacific region's uneven recovery and its status as one of the hardest hit regions (UNWTO, 2021).

Figures at the city and national levels show the shock and the different recovery paths of the three hubs under study. Amid border closures, Singapore's visitor arrivals dropped 85.7% to about 2.7 million in 2020 (from about 19.1 million in 2019) (Singapore Tourism Board [STB], 2021). In 2020, 3.57 million people visited Hong Kong (-93.6% year over year) (Hong Kong Tourism Board [HKTB], 2021). Thailand saw a sharp decline in visitors from almost 40 million in 2019 to about 6.7 million in 2020, with Bangkok serving as the main entry point (Reuters, 2025). A comparative multi-city forecasting design with an emphasis on Bangkok, Hong Kong, and Singapore is motivated by these indicators.

In order to help navigate post-pandemic volatility, the Smart Traveller Insights System, which is the project's proposed solution, is intended to serve as a decision support interface that displays short-horizon forecasts and model diagnostics. Conventional monitoring, which consists of purely descriptive dashboards and official statistics that are released slowly, are lagging indicators that find it difficult to support quick operational and policy decisions. Pre-pandemic tourism forecasting was largely based on univariate time-series models that assumed stable seasonality. However, structural breaks in 2020–2021 broke these assumptions, which frequently resulted in performance degradation unless models were supplemented with richer features or alternative approaches (Wang, Li, & Park, 2023; Li, 2021). Regularized linear methods and tree-based ensembles have been shown to outperform classical baselines when rigorously evaluated, highlighting the growing role of machine learning (ML) and Internet/search data in tourism demand forecasting (Li, 2021). In order to evaluate true value-add, this encourages combining contemporary machine learning models with an open seasonal naïve benchmark (Hyndman & Athanasopoulos, 2021).

Bangkok, Hong Kong, and Singapore combine the transparent, high-frequency data necessary for reliable modelling with the significant worldwide tourism influence. Bangkok is frequently ranked as one of the most visited cities in the world, both before and after the pandemic, highlighting its role as a bellwether (Mastercard, 2019; Euromonitor International, 2024). Consistent model training, validation, and cross-city comparison are made possible by the official monthly visitor-arrivals series that each of the three markets provides through their public portals: Thailand via the Bank of Thailand's tourism indicator tables, Hong Kong via HKTB/PartnerNet (which offers downloadable monthly reports), and Singapore via STB's analytics network (STB, 2021; HKTB, 2021; Bank of Thailand, 2025). While Kuala Lumpur was examined during the initial scoping phase, Bangkok was chosen as the Southeast Asia proxy due to data consistency and completeness, maintaining the regional perspective and bolstering methodological reliability.

1.2 PROBLEM BACKGROUND

1.2.1 Features of Post-Shock Data: Latency, Regime Shifts, and Structural Breaks

Following COVID-19, the Asia-Pacific region's monthly international arrivals series show level shifts, broken seasonality, and sporadic plateaus. These characteristics, which complicate short-horizon planning and disturb pre-2020 seasonal regularity, result from staggered border policies, airline capacity restoration, and changing risk perceptions (UNWTO, 2021). Furthermore, using static dashboards or descriptive bulletins only offers a limited amount of forward guidance because official releases are usually delayed. Forecasting on discontinuous series is therefore given precedence over retrospective description in a defensible analysis (UNWTO, 2021).

1.2.2 Pre-pandemic forecasting setup limitations under structural breaks

Prior to 2020, research on tourism demand frequently used univariate time-series models that assume stable seasonality, such as decomposable trend-seasonal forms or variants of ARIMA. These presumptions were regularly broken during and after COVID-19, and performance suffered unless models were enhanced with more detailed features or different learners (Wang, Li, & Park, 2023). According to recent reviews, machine learning and web/search data are becoming increasingly important in tourism forecasting. Regularized linear methods and tree-based ensembles, in particular, show promise in post-shock scenarios when assessed against open benchmarks (Li, 2021). In order to confirm true value-add

beyond merely seasonal repetition, this encourages the inclusion of a Seasonal Naïve comparator (Hyndman & Athanasopoulos, 2021).

1.2.3 Regular, Public Monthly Series Are Necessary for Cross-City Comparability

When demand composition, air connectivity, and policy cadence vary, conclusions drawn from a single market may be deceptive. This study chooses three hubs with regular, public monthly arrivals to allow for like-for-like comparison: Thailand (Bank of Thailand tourism indicators), Hong Kong (HKTB/PartnerNet; C&SD tables), and Singapore (STB analytics network). For consistent training and validation free from measurement artifacts from diverse collection methods, these sources offer the regularity and consistency needed (STB, 2021; HKTB, 2021; Bank of Thailand, 2025). Although Kuala Lumpur was examined during the initial scoping phase, Bangkok was chosen as the Southeast Asia proxy due to data consistency and completeness, maintaining the regional perspective while enhancing methodological reliability.

Plateaued Series Evaluation Practice

Variance-based measures can be deceptive on series that show long plateaus (such as periods of flat recovery). For the simple reason that test-set variance is low, a model may have low absolute error but poor R^2. In line with established forecasting practice and recent post-COVID methodological notes, this study prioritizes MAPE, MAE, and RMSE for primary assessment, while keeping R2 as a secondary diagnostic (Hyndman & Athanasopoulos, 2021; Wang et al., 2023).

1.3 PROBLEM AIM

To create a Smart Traveller Insights System that uses a consistent feature-engineered pipeline that evaluates Ridge Regression, Random Forest, and XGBoost against a Seasonal Naïve baseline, as measured by MAPE, MAE, RMSE, and R^2, to generate and visualize short-horizon forecasts of monthly international tourist arrivals for Singapore, Hong Kong, and Bangkok.

1.4 OBJECTIVES

O1. Dataset construction and comparability

To compile and record monthly international arrivals, Google Trends and Hotel Occupancy Rate datasets for Singapore, Hong Kong, and Bangkok that are consistent (2017–2024) while guaranteeing data quality, standardized definitions, and cross-city comparability.

O2: Design of forecasting framework

To create a comparative forecasting framework that combines data-driven predictive models with a transparent statistical baseline, backed by a consistent feature-engineering process and validation that is appropriate for time series.

O3. Cross-city analysis and quantitative assessment

To interpret the results across the three cities, assess forecast performance using pre-specified accuracy metrics (MAPE, MAE, RMSE, and R2), and determine the circumstances in which data-driven models outperform the baseline.

O4. Reproducibility and system realization

To create an interactive, research-grade dashboard with forecasts and diagnostics along with fully reproducible artifacts (code, data provenance, configuration) for the Smart Traveller Insights System.

1.5 SCOPE

1.5.1 Deliverables

Data collection: Compiling monthly data on foreign visitor arrivals from official portals for Singapore, Hong Kong, and Bangkok (a proxy for Thailand) from 2017–2024.

Data preprocessing: Cleaning, format reconciliation, missing value handling, and series alignment for uniform temporal coverage.

Feature engineering: Process of improving forecast ability by developing rolling statistics, calendar effects, and lag features.

Model Development: Forecasting monthly arrivals using XGBoost, Random Forest, Ridge Regression, and Seasonal Naïve training.

Model evaluation and optimization: choosing the best models for each city, documenting settings, and evaluating performance using MAPE, MAE, RMSE, and R2.

System Deployment: Using a Streamlit web application to package the forecasting process and display the outcomes.

Frontend development: the process of creating an interactive dashboard that shows forecasts, historical trends, and comparative diagnostics.

Reproducibility package: Code, environment files, configuration, and instructions to replicate results are all included.

1.5.2 Constraints

Revision risk and official data latency: All results reflect the most recent releases available at the time of analysis; monthly arrivals are released with lags and may be updated retroactively.

Variability in measurement: Even when one authoritative source is used per city, cross-jurisdictional differences (definitions, counting rules, special events) may still exist; this is recognized in the interpretation.

Scope of univariate forecasting: Only arrivals are modeled in the study; no exogenous regressors, such as air-seat capacity, exchange rates, or macro indicators, are included. The findings are not causative; they are predictive.

Properties of the post-shock series: Even when absolute errors are small, structural breaks, regime shifts, and plateaus can reduce variance-based metrics (like R^2) and limit the amount of accuracy that can be achieved.

Budgets for time and computation: The validation employs rolling/expanding-origin schemes suitable for a student project rather than large-scale experimentation, and the hyper-parameter search is bounded.

Model boundaries: Seasonal Naïve, Ridge, Random Forest, and XGBoost are the only forecasting frameworks compared; deep learning and probabilistic/quantile forecasting

frameworks are not included.

Limits of software and deployment: Authentication, load balancing, role-based access, and high availability hosting are not offered by the dashboard, which is a research-grade Streamlit application meant for academic demonstration.

Dependencies on reproducibility: The aforementioned Python environment, package versions, and public data portals must be available for reproduction; project control has no control over breaking API/portal changes.

Ethics and legal: There is no use of private information or scraping of restricted sources; only publicly available official statistics are used.

1.5.3 Project Inclusions

Time and place: Bangkok (Thailand proxy), Singapore, and Hong Kong; monthly frequency, January 2017–December 2024.

Data items: Arrivals series that have been cleaned; feature matrices that include calendar effects (month indicators), rolling summaries (e.g., 3/6-month means), and specific lags (e.g., 1, 3, 6, 12).

Forecast configuration: Monthly short-horizon forecasts (defined in Chapter 3 as operationally relevant horizons) generated by a standard pipeline for each city.

Set of models: Random Forest, XGBoost, Ridge Regression, and Seasonal Naïve (baseline).

Selection and validation: transparent model selection by city, time-series-appropriate validation (rolling/expanding origin), and informative ablation notes.

Metrics for evaluation: Error metrics in low-variance segments were given priority in the systematic reporting of MAPE, MAE, RMSE, and R^2.

Artefacts and outputs: Streamlit dashboard, per-city summaries, interpretive commentary,

comparative tables/plots, and a complete reproducibility bundle (code, environment files, configuration, instructions).

1.5.4 Project Exclusions

Other Cities: Due to data-consistency issues discovered during scoping, markets outside of the three hubs—such as Kuala Lumpur—are not included.

Granularity: that goes beyond the month. No breakdowns by origin-market, season/segment, daily, or sub-city.

Casual or exogenous drivers: No macroeconomic or epidemiological covariates, policy indices, flight capacity, fares, causal inference, or counterfactual analyses are included.

Model families are not taken into account: No probabilistic/quantile, LSTM/Transformer, ETS/Prophet, ARIMA/SARIMA, or hierarchical reconciliation techniques.

Quantification of uncertainty beyond the fundamentals: No calibrated prediction intervals or conformal bands are generated, except for diagnostics and basic error summaries.

Production: Testing is restricted to research-grade checks; there are no SLAs, CI/CD pipelines, container orchestration, autoscaling, or observability stacks.

Primary gathering of data: No experiments, interviews, or surveys.

Multi-platform or mobile applications: The interface is web-based; there are no native iOS or Android deliverables.

1.5.6 Targeted Users:

- **Tourism Boards:** Officials who require information on the well-being and confidence in recovery of foreign visitors.
- **Public health policymakers:** Urban health and safety planners looking for data-driven support to promote healthy cities.

- **Researchers and Analysts:** Academics with an interest in the relationship between public well-being, tourism, and digital behaviour.
- **Travellers in general:** Those who might use the Traveller Confidence Index to evaluate the safety of their trip and make appropriate plans.

1.6 POTENTIAL BENEFITS

1.6.1 Tangible Benefits

Forecasts on coherent monthly series that are comparable: Using official series, a uniform pipeline provides structured evidence on post-shock recovery dynamics, producing monthly forecasts for Bangkok, Hong Kong, and Singapore that are like-for-like (UNWTO, 2021).

Measured improvements over a basic standard: Improvements are auditable and attributable through systematic evaluation against a Seasonal Naïve comparator, which is in line with accepted forecasting pedagogy (Hyndman & Athanasopoulos, 2021).

Rigorous evaluation in the spirit of competition: The results' credibility is increased by transparent selection and pre-declared accuracy metrics (MAPE, MAE, RMSE, and R2) that reflect best practices from extensive forecasting studies (Makridakis, Spiliotis, & Assimakopoulos, 2022).

Research artifacts that can be replicated: Replication and extension in coursework and dissertations are made possible by public code, environment files, and documented splits (Li, 2021).

1.6.2 Intangible Benefits

Contribution of methodology in post-shock situations: The post-COVID tourism forecasting discourse is bolstered by evidence of when regularized linear models and tree ensembles perform better than seasonal repetition on series with broken seasonality and plateaus (Li, 2021).

Educational worth: In line with commonly taught forecasting guidance, a baseline-first design upholds fundamental principles (the need for naïve comparators; cautious metric

selection when variance is low) (Hyndman & Athanasopoulos, 2021).

Conformity to SDG 3: Good Health and Well-Being. The study contributes to the body of knowledge related to SDG 3's focus on resilient health and well-being by organizing prompt, evidence-based recovery analysis for international travel systems interrupted by a health emergency (United Nations, n.d.; UNWTO, 2021).

1.7 Overview of The IR

Chapter 1 (The Introduction). The post-pandemic context for the Asia-Pacific tourism recovery is outlined in this chapter, which also serves as the impetus for a forecasting-centered analysis of three urban centers: Bangkok, Hong Kong, and Singapore. It outlines four outcome-oriented objectives, a single, clear project aim, and the work's tangible and intangible benefits. It also defines the scope by clearly defining deliverables, constraints, inclusions, exclusions, and the intended user. The chapter provides the theoretical and practical framework that the rest of the report is built around.
Chapter 2: Review of Literature. Instead of offering a descriptive catalog, this chapter offers a critical synthesis. Prior to reviewing technical research pertinent to a multi-city monthly forecasting pipeline (data handling, feature engineering, model classes, and evaluation practice), it places the project within domain research on post-shock tourism dynamics and measurement. It then examines related works to identify methodological strengths, limitations, and transferable design choices. The chapter ends by pointing out particular gaps that the project fills and by drawing conclusions about design that guide the analytical strategy that was selected.

Methodology, Chapter 3. This chapter maps the project's phases to scheduled activities and provides justification for the overall development lifecycle that was chosen. It outlines the feature-engineering protocol, formalizes data acquisition and preprocessing methods, and describes the comparative forecasting design that includes machine-learning families and a transparent statistical baseline. In order to prevent leakage and overfitting, it also lays out the validation plan and pre-stated evaluation metrics. The chapter acts as an operational implementation blueprint, coordinating methodological decisions with the goals outlined in Chapter 1 and the gaps mentioned in Chapter 2.

Design and Implementation, Chapter 4. The Smart Traveller Insights System's realization will be documented in this chapter, which was finished during the project's second phase. The methodological blueprint will be translated into tangible artifacts, such as curated dictionaries and datasets, feature-engineering scripts that provide explanations for calendar, rolling, and lag constructs, baseline and machine-learning models that are implemented with training logs and configuration records, and the web-based interface that displays comparative forecasts and diagnostics.

Results and Discussion, Chapter 5. The empirical evaluation will be reported in this chapter while closely adhering to the stated protocols. In order to explain relative performance under post-shock, plateau-prone regimes, it will provide accuracy results (MAPE, MAE, RMSE, and R2) for every city along with residual analyses and diagnostic plots. Cross-city interpretation will highlight comparability, relate results to the Chapter 2 review, and define the boundaries of inference within the specified temporal and geographic scope.

Chapter 6: Conclusion and Upcoming Projects. The project's contributions and limitations in relation to the stated aim and objectives will be summarized in the final chapter, which will also consider the methodological and practical implications of the findings and suggest feasible extensions. I

1.7 Project Plan

1.1 Project Plan

Task Name	Duration	Start Date	End Date	Status
Project Proposal Form	7 days	24/02/2025	03/03/2025	Complete
Chapter 1: Introduction	**3 days**	**29/03/2025**	**31/03/2025**	**Complete**
1.1 Introduction	<1 day	29/03/2025	29/03/2025	Complete
1.2 Problem Background	<1 day	29/03/2025	29/03/2025	Complete
1.3 Project Aim	<1 day	29/03/2025	29/03/2025	Complete
1.4 Objectives	<1 day	30/03/2025	30/03/2025	Complete
1.5 Scope	<1 day	30/03/2025	30/03/2025	Complete
1.6 Potential Benefits	<1 day	31/03/2025	31/03/2025	Complete
1.7 Overview of the IR	<1 day	31/03/2025	31/03/2025	Complete
1.8 Project Plan	<1 day	31/03/2025	31/03/2025	Complete
Chapter 2: Literature Review	**8 days**	**04/04/2025**	**13/04/2025**	**Complete**
2.1 Introduction	1 day	04/04/2025	04/04/2025	Complete
2.2 Domain Research	3 days	05/04/2025	08/04/2025	Complete
2.3 Similar Systems/Works	2 days	09/04/2025	10/04/2025	Complete
2.4 Technical Research	1 day	12/04/2025	12/04/2025	Complete
2.5 Summary	<1 day	13/04/2025	13/04/2025	Complete
Chapter 3: Methodology	**9 days**	**15/04/2025**	**29/04/2025**	**Complete**
3.1 Introduction	1 day	15/04/2025	15/04/2025	Complete
3.2 Methodology	2 days	17/04/2025	19/04/2025	Complete
3.3 Data Collection and Data Understanding	3 days	22/04/2025	25/04/2025	Complete
3.4 Initial Data Preprocessing	2 days	26/04/2025	28/04/2025	Complete
3.5 Summary	1 day	29/04/2025	29/04/2025	Complete
Chapter 4: CONCLUSION	**<1 day**	**01/05/2025**	**01/05/2025**	**Complete**
References	**<1 day**	**01/05/2025**	**01/05/2025**	**Complete**
Appendices	**<1 day**	**01/05/2025**	**01/05/2025**	**Complete**

Table 1 Project Plan for Semester 1

Task Name	Duration	Start Date	End Date	Status
Project Proposal Form	7 days	24/02/2025	03/03/2025	Complete

Chapter 1: Introduction	**3 days**	**29/03/2025**	**31/03/2025**	**Complete**
1.1 Introduction	<1 day	29/03/2025	29/03/2025	Complete
1.2 Problem Background	<1 day	29/03/2025	29/03/2025	Complete
1.3 Project Aim	<1 day	29/03/2025	29/03/2025	Complete
1.4 Objectives	<1 day	30/03/2025	30/03/2025	Complete
1.5 Scope	<1 day	30/03/2025	30/03/2025	Complete
1.6 Potential Benefits	<1 day	31/03/2025	31/03/2025	Complete
1.7 Overview of the IR	<1 day	31/03/2025	31/03/2025	Complete
1.8 Project Plan	<1 day	31/03/2025	31/03/2025	Complete
Chapter 2: Literature Review	**8 days**	**04/04/2025**	**13/04/2025**	**Complete**
2.1 Introduction	1 day	04/04/2025	04/04/2025	Complete
2.2 Domain Research	3 days	05/04/2025	08/04/2025	Complete
2.3 Similar Systems/Works	2 days	09/04/2025	10/04/2025	Complete
2.4 Technical Research	1 day	12/04/2025	12/04/2025	Complete
2.5 Summary	<1 day	13/04/2025	13/04/2025	Complete
Chapter 3: Methodology	**9 days**	**15/04/2025**	**29/04/2025**	**Complete**
3.1 Introduction	1 day	15/04/2025	15/04/2025	Complete
3.2 Methodology	2 days	17/04/2025	19/04/2025	Complete
3.3 Data Collection and Data Understanding	3 days	22/04/2025	25/04/2025	Complete
3.4 Initial Data Preprocessing	2 days	26/04/2025	28/04/2025	Complete
3.5 Summary	1 day	29/04/2025	29/04/2025	Complete
Chapter 4: CONCLUSION	**<1 day**	**01/05/2025**	**01/05/2025**	**Complete**
4.1 Introduction	1 Day	Wed, 20/08/25	Wed, 20/08/25	Completed
4.2 Data Collection	1 Day	Wed, 20/08/25	Wed, 20/08/25	Completed
4.3 Data Understanding	1 Day	Thurs, 21/08/25	Thurs, 21/08/25	Completed
4.4 Data Pre-processing	2 Days	Fri, 22/08/25	Sat, 23/08/25	Completed
4.5 Model Building	8 Days	Sun, 24/08/25	Sun, 31/08/25	Completed

4.6 Summary	1 Day	Sun, 31/08/25	Sun, 31/08/25	Completed
CHAPTER 5: RESULTS AND DISCUSSIONS	**12 Days**	**Mon, 1/09/25**	**Fri, 12/09/25**	**Completed**
5.1 Introduction	1 Day	Mon, 1/09/25	Mon, 1/09/25	Completed
5.2 Model Evaluations and Discussions	5 Days	Tue, 2/09/25	Sat, 6/09/25	Completed
5.3 Model Deployment	6 Days	Sun, 7/09/25	Fri, 12/09/25	Completed
5.4 Summary	1 Day	Fri, 12/09/25	Fri, 12/09/25	Completed
CHAPTER 6: CONCLUSION	**4 Days**	**Sat, 13/09/25**	**Tue, 16/09/25**	**Completed**
6.1 Critical Evaluation	4 Days	Sat, 13/09/25	Tue, 16/09/25	Completed
6.2 Limitations	4 Days	Sat, 13/09/25	Tue, 16/09/25	Completed
6.3 Recommendations	4 Days	Sat, 13/09/25	Tue, 16/09/25	Completed
References	**<1 day**	**01/05/2025**	**01/05/2025**	**Complete**
Appendices	**<1 day**	**01/05/2025**	**01/05/2025**	**Complete**

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

Following the COVID-19 pandemic, there has been a significant methodological change in the forecasting of tourism demand. Econometric and univariate time-series models, such as ARIMA, SARIMA, and ETS, were primarily used in pre-pandemic research. These models performed well in situations with steady seasonality and slow growth trends (Li, Song, & Witt, 2005). These models' vulnerability was brought to light by the abrupt structural disruptions and regime changes brought about by the pandemic, as their dependence on stationarity and recurring seasonal cycles was no longer valid (Qiu, Wu, Dropsy, Petit, & Pratt, 2021). This has sparked a new wave of research that uses hybrid forecasting frameworks, machine learning (ML), and ensemble learners to capture irregular fluctuations and nonlinear recovery dynamics (Kumar, Misra, & Chan, 2022; Yang, Li, Guo, & Sun, 2023).

There is increasing evidence that machine learning models like Ridge regression, Random Forest, and XGBoost can perform better than traditional benchmarks when tested in unstable, post-shock environments (Kayral, Sarı, & Aktepe, 2023; Lee, 2025). Comparability between studies is limited because these techniques are not regularly evaluated against clear baselines like Seasonal Naïve. The majority of post-pandemic research is still concentrated on forecasting at the country or single destination level, which is another gap that has been noted (Polyzos, Fotiadis, & Samitas, 2021). This raises questions about how recovery paths vary among globally important Asia-Pacific hubs, where traveller confidence, air connectivity, and border policies vary. Additionally, although a number of studies incorporate web search data, mobility indicators, or decomposition frameworks, little progress has been made in creating repeatable pipelines that combine interpretability, practical deployment, and predictive accuracy (Chen, Hu, & Law, 2025; Li, Yang, Guo, Sun, & Wang, 2024).

2.2 Domain Research

2.2.1 Tourism Demand Forecasting Pre-Pandemic (Classical Foundations)

Prior to the COVID-19 pandemic, time-series and classical econometric models dominated the forecasting of tourism demand. Because they provided interpretability and demonstrated consistent performance under stable seasonal conditions, methods like ARIMA, SARIMA, Holt–Winters exponential smoothing, and ETS were widely used. Their success was based on the presumption of gradual growth and recurrent seasonality, which matched historical trends in foreign arrivals.

However, the pandemic brought to light their shortcomings. These models were not built to handle shocks like the abrupt decline in international mobility, extended travel restrictions, and non-linear recovery trajectories. Recent research indicates that when applied to post-2020 tourism data, forecasts based on classical approaches showed systematic biases and were unable to adjust to abrupt structural breaks (Qiu, Wu, Dropsy, Petit, & Pratt, 2021). This reaffirmed the worry that traditional models are not resilient in situations of increased volatility, even though they are still useful for benchmarking.

Simple statistical comparators are still useful in tourism forecasting research in spite of these drawbacks. When assessing whether machine learning techniques actually add value, transparent baselines like Seasonal Naïve offer a crucial benchmark against which more advanced approaches can be assessed (Li, Yang, Guo, Sun, & Wang, 2024). Maintaining the use of these baselines prevents the tendency to exaggerate the advantages of sophisticated models and guarantees methodological rigor.

In conclusion, traditional forecasting techniques laid the groundwork for modelling tourism demand and are still useful as clear standards. However, the need for more adaptable strategies is highlighted by their incapacity to adjust to structural shocks. In order to better understand post-pandemic volatility and recovery dynamics, the current study advances machine learning models while maintaining Seasonal Naïve as a baseline comparator.

2.2.2 Post-COVID Tourism Dynamics and Structural Breaks

The COVID-19 pandemic caused a previously unheard-of disruption in global travel patterns, which led to circumstances that seriously questioned conventional forecasting techniques. Global tourist arrivals fell by over 80% in 2020 as a result of travel restrictions, border closures, and lockdowns; the Asia-Pacific region saw the biggest drop (Qiu, Wu, Dropsy, Petit, & Pratt, 2021). In contrast to earlier disruptions like natural disasters or financial crises, the pandemic's effects were systemic and prolonged, causing demand patterns to structurally break as well as temporarily decline. Long-term patterns were upset, seasonal peaks vanished, and recovery paths became significantly uneven across locations (Polyzos, Fotiadis, & Samitas, 2021).

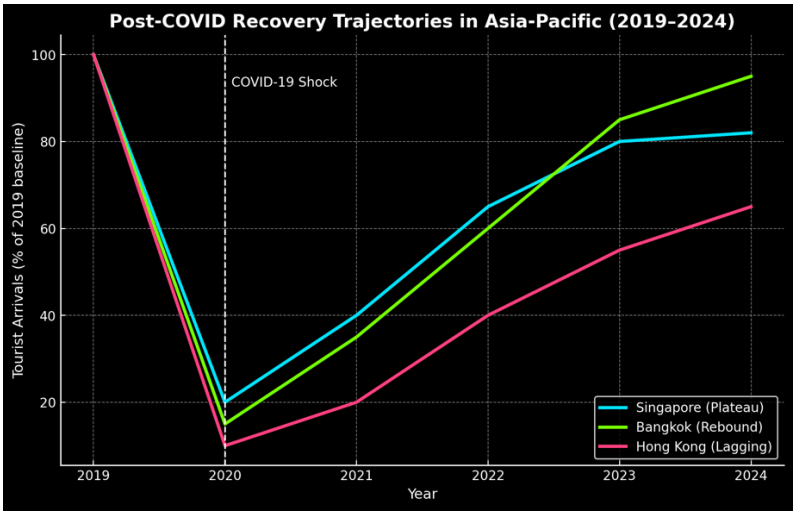


Figure 1 Image depicting post covid trajectories in arrivals for Singapore, Hong Kong and Bangkok

The image illustrated showcases how covid impacted these three Asia pacific hubs and how the covid shock impacted them in terms of tourism. Singapore and Hong Kong depict low recovery while Bangkok seems to have almost reached full recovery.

Recovery has not been linear and has depended on geography, according to recent studies. While hubs heavily reliant on long-haul or Mainland Chinese Travellers trailed behind, destinations with diverse source markets, robust domestic tourism bases, or quicker vaccine rollouts showed earlier rebounds (Yang, Li, Guo, & Sun, 2023). For example, Bangkok's demand was more erratic but regained seasonal momentum once restrictions were loosened, whereas Singapore's cautious approach to border reopening resulted in a plateaued recovery path. Geopolitical factors and stringent zero-COVID policies, on the other hand, caused Hong Kong to experience protracted stagnation. These varying results highlight the shortcomings of

general forecasting and the requirement for city-specific models that can take into consideration different recovery dynamics.

The growing significance of uncertainty and Traveller confidence has been another characteristic of the post-pandemic environment. Indicators like mobility data, web search activity, and policy stringency indices have shown promise in capturing changes in travel intentions and constraints, adding value beyond raw arrival figures (Li, Yang, Guo, Sun, & Wang, 2024; Chen, Hu, & Law, 2025). This supports the claim that forecasting in the post-COVID era needs to incorporate exogenous variables or machine learning architectures that can handle sudden regime changes, in addition to traditional seasonality-based models.

Three conclusions can be drawn from the literature as a whole. First, COVID-19 was a structural shock that disproved the stationarity assumptions that underlie traditional forecasting techniques, rather than just a transient disruption. Second, recovery in Asia-Pacific locations has been uneven, which calls for comparative methods that can highlight resilience in individual cities. Third, a move toward data-driven and adaptable frameworks is indicated by the increasing significance of behavioural and high-frequency indicators. The present study, which uses Singapore, Bangkok, and Hong Kong as representative examples of diverse post-COVID tourism recovery, is empirically supported by these findings.

2.2.3 Machine Learning and Hybrid Forecasting Methods

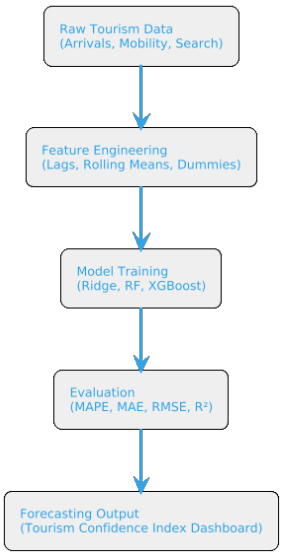


Figure 2 Image depicting ML flow work for this project

One of the biggest methodological changes in the tourism industry is the move to machine learning (ML) in demand forecasting. ML techniques are data-driven, non-parametric, and flexible, which makes them more robust in the face of volatility and structural breaks than traditional econometric and time-series models, which depend on linearity, stationarity, and seasonal regularity. The COVID-19 pandemic hastened this shift by highlighting the shortcomings of conventional methods and generating a need for models that can manage high-dimensional, non-linear, and dynamic data environments (Kayral, Sarı, & Aktepe, 2023; Li, Yang, Guo, Sun, & Wang, 2024).

Algorithms that extract intricate patterns from data without predetermined functional forms are commonly referred to as machine learning techniques. This enables forecasting contexts to model interactions between lagged features, capture nonlinear dependencies, and adjust to asymmetric recovery dynamics that surfaced in post-pandemic tourism flows. By minimizing loss functions under regularization constraints, tree ensembles, or iterative boosting mechanisms, machine learning models maximize prediction accuracy in contrast to regression-based econometrics, which necessitates strong assumptions about error structures and trend behavior. Because of their adaptability, they are especially well-suited for post-shock, short-horizon forecasting, where conventional assumptions are no longer valid (Chen, Hu, & Law, 2025).

This change is demonstrated by recent applications in forecasting tourism demand. According to studies, Random Forest and XGBoost perform better than ARIMA and ETS at forecasting erratic post-COVID visitor arrivals, especially when exogenous data like web search indices, rolling averages, and lag features are included (Lee, 2025). Ridge regression has also demonstrated efficacy in high-dimensional contexts, stabilizing estimates in the presence of seasonal dummies and multiple lag structures (Li et al., 2024). By separating trend, seasonal, and irregular components prior to modelling residual variance, hybrid frameworks—which integrate decomposition techniques with machine learning learners—further improve adaptability. When taken as a whole, these strategies show that machine learning is a methodological response to the new forecasting difficulties brought about by COVID-19, rather than just a more complicated substitute.

Nevertheless, there are some difficulties in implementing ML in tourism forecasting. While ensemble methods like XGBoost provide high accuracy, their black-box nature raises concerns about transparency in policymaking contexts, which is a common criticism in the literature (Chen et al., 2025). Furthermore, it is challenging to evaluate the true incremental value of sophisticated techniques since many ML applications lack systematic benchmarking against straightforward but effective baselines like Seasonal Naïve. Another issue is reproducibility, which is frequently hampered by the intricacy of data pipelines and hyperparameter tuning (Kayral et al., 2023). These drawbacks highlight how crucial it is to incorporate machine learning techniques into organized, replicable frameworks that blend methodological transparency with predictive power.

This domain explores the mathematical and applied underpinnings of three chosen models: Ridge regression, Random Forest, and XGBoost, in order to situate the current study within this developing corpus of work. The theoretical foundations, applicability for post-shock tourism forecasting, and empirical performance in recent studies are all taken into consideration when evaluating each model. In order to bridge the gap between traditional interpretability and machine learning flexibility, hybrid approaches and decomposition frameworks are also taken into consideration. These techniques work together to form the analytical foundation of the Smart Traveller Insights System, guaranteeing transparent benchmarking against baseline models as well as predictive robustness.

Regularized Regression: Ridge Regression

For high-dimensional forecasting scenarios, ridge regression is a straightforward yet effective modification of linear regression. Ridge provides stability by reducing coefficient estimates in tourism forecasting, where multicollinearity is frequently produced by lag features, seasonal dummies, and rolling averages. The function of optimization is:

$$\hat{\beta} = \arg\min_{\beta} \left(\sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right)$$

Equation 1 Ridge Regression Formula

Here, λ regulates the degree of penalization: the model reduces to ordinary least squares

(OLS) when $\lambda = 0$, and coefficients shrink towards zero as $\lambda \to \infty$ $\lambda \to \infty$. This guarantees reliable parameter estimation even in the presence of correlated predictors.

Application in Tourism: When modelling daily tourist arrivals during unstable COVID-19 recovery phases in China, Li et al. (2024) showed that Ridge regression performs better than ARIMA. Ridge provided consistent, comprehensible forecasts by combining several lagged predictors.

Random Forest Ensemble Learning

Random Forest (RF) reduces variance and increases predictive accuracy by combining multiple decision trees using bootstrap aggregation, also known as bagging. Its function for making predictions is:

$$\hat{y} = \frac{1}{B} \sum_{b=1}^{B} T_b(x)$$

Equation 2 Random Forest Formula

where Tb(x) is the b-th tree's prediction. At each split, a random subset of predictors is taken into consideration, and each tree is trained on a bootstrap sample.

Benefits:

- captures nonlinearities without any predetermined forms.
- resistant to overfitting and noise.
- Models feature interactions naturally.

Applications in Tourism: When predicting Turkey's post-COVID arrivals, Kayral et al. (2023) discovered that RF performed better than SARIMA, especially during times of erratic seasonal rebound.

XGBoost for Gradient Boosting

The gradient boosting technique XGBoost builds trees one after the other, fixing mistakes from previous iterations. Its goal is to:

$$Obj^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t)$$

Equation 3 XGboost Formula

where $\Omega(fk)$ penalizes model complexity to avoid overfitting, and l is the loss function (such as squared error).

Benefits:

- Extremely precise in dynamic recovery conditions.
- Incorporates stability through regularization (λ, γ).
- Manages sparsely featured heterogeneous data.

Applications in Tourism: Chen, Hu, and Law (2025) combined XGBoost with multi-channel data (weather, search indices, and visitor arrivals) and outperformed ARIMA and Prophet in post-pandemic recovery forecasts.

Decomposition and Hybrid Methods

ML algorithms and statistical decomposition are combined in hybrid models. An example of a typical workflow:

- Break down time series into irregular, seasonal, and trend components.
- Apply machine learning techniques (such as RF, XGBoost, and LSTM) to irregular terms or residuals.
- Reassemble forecasts by reassembling their constituent parts.

Applications in Tourism: Li et al. (2024) used a decomposition-ensemble model that greatly increased the accuracy of daily tourism demand forecasting, demonstrating that hybrids can perform better than standalone machine learning and traditional methods.

Measures of Evaluation

To guarantee robustness, ML models are usually assessed using a variety of error metrics:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \qquad MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \qquad R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

RMSE: Sensitive to significant mistakes.

MAE: Sturdy against anomalies.

MAPE: Widely used after COVID, useful in a range of magnitudes.

R2: Frequently deceptive in recovery series that have plateaued.

Comparative Evidence

Study	Models Tested	Dataset	Result	Limitation
Qiu et al. (2021)	ARIMA vs SARIMA	Asia-Pacific arrivals 2020–21	Poor under shocks; large error margins	Could not adapt to structural breaks
Kayral et al. (2023)	ANN, RF, SARIMA	Turkey arrivals (monthly)	RF & ANN > SARIMA during volatile recovery	Limited cross-destination analysis
Li et al. (2024)	Hybrid decomposition	China arrivals (daily)	Hybrid > ARIMA, Ridge, standalone ML	Computationally intensive
Lee (2025)	SARIMAX + search data	East Asia tourist flows	Web data integration improved short-horizon accuracy	Requires high-frequency external data
Chen et al. (2025)	XGBoost, Prophet, ARIMA	Multi-source, post-COVID	XGBoost achieved highest predictive accuracy	Interpretability remains limited

Table 2 Comparison of the models through previous works

Synthesis

The literature shows that by capturing nonlinear dependencies, stabilizing estimates under multicollinearity, and modelling abrupt post-COVID structural breaks, machine learning techniques like Edge Regression, Random Forest, and XGBoost address major flaws in traditional tourism forecasting. Despite having a higher computational cost, hybrid decomposition further improves adaptability. Notwithstanding these benefits, there are still issues with reproducibility, benchmarking, and interpretability. In order to ensure methodological transparency and relevance to Asia-Pacific post-pandemic recovery, the current study systematically compares Ridge, Random Forest, and XGBoost against Seasonal Naïve within a reproducible pipeline.

2.2.4 Practices for Cross-City Forecasting and Assessment

Forecasting Across Cities in Tourism Studies

Historically, the literature on tourism forecasting has been destination-specific, concentrating on individual markets like Chinese domestic flows, inbound arrivals to Spain, or particular island economies. The different recovery paths that developed across locations in the wake of COVID-19 are not captured by these studies, despite the fact that they provide methodological insights. Due to exposure to geopolitical constraints, reliance on foreign versus domestic markets, and variations in policy interventions, recovery patterns have been extremely uneven (Polyzos, Fotiadis, & Samitas, 2021; Yang, Li, Guo, & Sun, 2023).

For instance, Hong Kong trailed behind because of extended zero-COVID restrictions, Bangkok showed an erratic recovery as regional mobility resumed, and Singapore's recovery plateaued as a result of its cautious border reopening strategy. There aren't many comparative forecasting frameworks in the literature, despite these glaring differences. Because comparative forecasting not only increases model generalizability but also identifies the structural drivers of resilience and vulnerability across destinations, the lack of systematic cross-city analyses represents a significant gap (Li, Yang, Guo, Sun, & Wang, 2024).
By concentrating on three Asia-Pacific hubs—Singapore, Bangkok, and Hong Kong—selected for their global connectivity, consistent data availability, and divergent recovery patterns, the current study fills this knowledge gap. In keeping with recent calls for comparative and multi-market forecasting frameworks in post-pandemic tourism research, this cross-city design offers a foundation for comparing model performance under diverse recovery conditions (Chen, Hu, & Law, 2025).

Post-COVID Forecasting Assessment Frameworks

Although statistical error metrics are frequently used to discuss forecasting accuracy, recent research highlights the significance of evaluation design in assessing the robustness of findings. In volatile recovery settings, where the data-generating process experiences structural breaks, traditional fixed train-test splits are frequently insufficient. To better capture the dynamic nature of recovery, rolling-origin and expanding-window evaluations have been suggested (Qiu, Wu, Dropsy, Petit, & Pratt, 2021).

Workflow for Rolling-Origin Evaluation

Models are repeatedly trained on increasingly updated training sets and assessed on shifting forecast horizons using the rolling-origin technique. This simulates forecasting conditions in the real world, where models are constantly re-estimated in response to new data. This procedure is depicted in the figure below.

This approach provides a more accurate evaluation of model robustness by generating a distribution of forecast errors across several origins. In order to stabilize performance under erratic trajectories, expanding windows—where the training set expands while the test horizon stays constant—are also helpful.

Methods of Comparative Assessment

The necessity of transparent and consistent error metrics across destinations is another recurrent theme in recent literature. Even though RMSE, MAE, MAPE, and R2 are still standard, their meanings change in post-COVID environments. For example, because MAPE expresses accuracy in percentage terms, it is especially well-suited to uneven recovery series. In contrast, R2 can yield misleading values when variance is low, as was the case with Singapore's plateaued recovery. Therefore, MAPE and MAE are emphasized as primary indicators in comparative studies, while R2 is kept as a secondary diagnostic and RMSE is used to penalize large deviations (Li et al., 2024; Chen et al., 2025).

Comparative Evidence from Literature

Study	Destinations	Models	Evaluation Design	Key Findings
Qiu et al. (2021)	Asia-Pacific (multi)	ARIMA, SARIMA	Fixed splits	Poor under shocks; large error margins
Polyzos et al. (2021)	ASEAN + East Asia	Econometrics	Scenario comparisons	Strong cross-country asymmetries in recovery
Li et al. (2024)	China (daily flows)	Hybrid ML + Decomposition	Rolling-origin	Hybrid improved short-horizon accuracy; robust design

Yang et al. (2023)	East Asia (regional)	Deep learning ensemble	Spatiotemporal evaluation	Highlighted inter-city spillover effects
Chen et al. (2025)	Multi-destination	XGBoost, Prophet, ARIMA	Expanding windows + Naïve baseline	XGBoost superior, but Seasonal Naïve critical baseline

Table 3 Comparison evidence of previous works with key findings

Synthesis

The majority of the contributions to the scant literature on cross-city tourism forecasting concentrate on individual destinations. Significant regional asymmetries are highlighted by the few multi-destination studies, highlighting the value of comparative methods. The literature also emphasizes that transparent design decisions—particularly rolling-origin evaluations and consistent baselines—are essential for robust evaluation, which calls for more than just error metrics.

By (i) performing a multi-city comparative analysis of Singapore, Bangkok, and Hong Kong, (ii) using a rolling-origin evaluation design, and (iii) giving preference to error metrics appropriate for post-COVID volatility (MAPE, MAE, RMSE) over potentially deceptive indicators like R2, this study advances this developing field.

2.2.5 Decision-Support Systems and Their Useful Consequences

A persistent problem in tourism research is converting forecasting results into formats that analysts, industry stakeholders, and policymakers can use. No matter how sophisticated the statistics, forecasting models are useless if the results are not clear, understandable, and useful. Decision-support systems (DSS), which incorporate predictive analytics into dashboards, indices, and visualization tools, have become increasingly important in the tourism industry as a result of this recognition (Gretzel, Werthner, Koo, & Lamsfus, 2020). The COVID-19 pandemic highlighted the necessity of prompt, open, and adaptable forecast communication in the face of unstable recovery conditions, underscoring the urgency of such systems.

Indicators of Confidence and Tourism Recovery

A popular strategy for succinctly expressing complex tourism dynamics is the use of composite indices. For example, tourism recovery indices combine data from several indicators, including flight capacity, international arrivals, search patterns, and policy

restrictions, to create a single measure that shows how well a destination's tourism industry is doing overall (UNWTO, 2022). There are two main functions of these indices:
Simplifying means turning complex dynamics into signals that can be understood.
Comparability allows for benchmarking between locations and eras.
Their function during the COVID-19 recovery is highlighted by empirical research. In order to provide tourism boards with useful information, Liu, Xu, and Li (2023) built a composite dashboard for Southeast Asia that integrated flight, hotel, and mobility data into a regional tourism index. Similarly, Song, Qiu, and Park (2021) contend that by offering an intuitive indicator of recovery momentum, these indices improve policy responsiveness.

Visualization and Interactive Dashboards

In tandem with indices, interactive dashboards have taken center stage in forecast distribution. Multiple model integration, user-driven scenario exploration, and real-time visualization of predictive outputs are made possible by tools like Streamlit, Tableau, and Power BI. Interactivity is a significant benefit since it allows non-technical users to observe forecast uncertainty, compare baseline and advanced models, and test scenarios (Shahrabi, Rajabifard, & Kalantari, 2022).

Applications for tourism have included everything from destination-specific decision tools that integrate web search and mobility data to regional COVID-19 monitoring dashboards. Nguyen, Pham, and Tran (2022), for instance, developed a dashboard for Vietnam's tourism ministry that allowed for evidence-based decision-making by visualizing recovery trajectories under various reopening strategies.

DSS Difficulties for Tourism Prediction

Notwithstanding these developments, the literature notes enduring difficulties in creating a successful DSS:

- **Complexity vs. interpretability:** While ML-based dashboards may seem confusing to non-technical users, composite indices run the risk of being overly simplistic (Chen, Hu, & Law, 2025).
- **Data integration:** A lot of systems rely on timely external data (such as policy indices, mobility, and search data), which might not always be accessible in different locations.

- **Reproducibility:** The lack of clear methodological documentation in many systems restricts their applicability in different contexts (Shahrabi et al., 2022).

These problems show how crucial it is to balance interpretability, accuracy, and transparency when designing decision-support tools.

Synthesis and Pertinence to the Current Research

According to the reviewed literature, DSS has become essential in the post-pandemic tourism landscape thanks to indices and dashboards. However, there are still gaps: the majority of systems either lack transparent benchmarking against basic statistical baselines or have a narrow focus on a single destination. Moreover, very few integrate comparative, multi-city recovery analysis with strong machine learning predictions.

By creating the Tourism Confidence Index (TCI), a Streamlit-based dashboard that combines XGBoost, Random Forest, and Ridge Regression forecasts for Singapore, Bangkok, and Hong Kong, the current study fills this gap.

The purpose of the TCI is to:

- Deliver measurable, transparent results that are compared to Seasonal Naïve.
- Integrate intuitive indices with error diagnostics to improve interpretability.
- Make it easier to compare locations with different recovery routes.

This guarantees that forecasting results are in line with new best practices in the literature and are both methodologically sound and useful in a decision-support setting.

2.3 Similar Systems

Research Topic	**Author(s)**	**Description**	**Dataset Used**	**Analysis Technique**	**Evaluation Result**	**Strength/Outcome**	**Limitation**
Global Tourism Recovery Monitoring	UNWTO (2022)	Developed the *Tourism Recovery Tracker*	Monthly international arrivals	Composite index methodology; descriptive	Not reported.	Enabled policymakers to benchmark recovery	Purely descriptive; no forecasting; aggregation

		integrating arrivals, flight capacity, and mobility indicators into a global monitoring platform. Provides cross-country comparability.	(UNWTO), flight seat capacity (IATA), mobility reports (Google, Apple), 2019–2022.	integration of heterogeneous datasets. No predictive modelling.		speed globally. First comprehensive, public-facing system post-COVID.	masks intra-regional heterogeneity; methodology opaque.
Composite Recovery Index (Southeast Asia)	Liu, Xu, & Li (2023)	Proposed a tourism recovery index tailored for Southeast Asia, combining aviation, hotel, and mobility data to monitor rebound.	Daily flight capacity, hotel occupancy, and mobility data across Southeast Asia (2019–2022).	Weighted composite index constructed with PCA validation. Integrated time series normalization and scaling for comparability.	RMSE ≈ 9.8%; MAPE ≈ 7.2%.	Produced real-time, interpretable signals; allowed destination benchmarking; robustly validated with PCA loadings.	Regional focus only; aggregation risked bias from dominant markets (e.g., Singapore, Thailand); no predictive capability.
National Recovery Dashboard (Vietnam)	Nguyen, Pham, & Tran (2022)	Built an interactive dashboard for	Monthly arrivals, government	ARIMA models applied to monthly arrivals;	RMSE ≈ 12.4%; MAPE ≈ 9.1%.	Dashboard allowed policymakers to test "what-	ARIMA underperformed during volatile COVID waves;

		Vietnam's ministry to simulate recovery trajectories under reopening policies.	policy stringency index, Google mobility indicators (2019–2021).	scenario-based simulation (optimistic, moderate, conservative reopening). Rolling-window validation used.		if" reopening strategies; provided scenario-based visualizations.	system lacked multi-country applicability; limited methodological transparency.
Hybrid Forecasting Framework (China)	Li, Yang, Guo, Sun, & Wang (2024)	Designed a hybrid forecasting approach integrating signal decomposition and machine learning to improve short-term accuracy.	Daily tourist arrivals across multiple Chinese cities (2018–2022).	CEEMDAN decomposition used to isolate components. Ridge regression, Random Forest, and XGBoost modeled residuals. Predictions ensembled via weighted voting. Rolling-origin validation benchmarked against ARIMA.	MAPE ≈ 6.5%; RMSE reduced by 18% vs ARIMA.	Demonstrated superiority of decomposition–ensemble models in capturing nonlinear shocks; robustness under structural breaks; significant accuracy improvements.	High computational demands; framework limited to Chinese datasets; no visualization or deployment system.
Multi-Channel Imaging	Chen, Hu, & Law (2025)	Developed a forecasting	Monthly arrivals, Google	CNN-based imaging models	MAPE ≈ 5.1%; RMSE	Achieved state-of-the-art accuracy;	Low interpretability; complex

Model (Asia-Pacific)		framework converting arrivals, search, and weather data into multi-channel images for deep learning.	Trends, meteorological data across Asia-Pacific (2019–2023).	combined with XGBoost and Prophet. Hyperparameters tuned via grid search. Benchmarked against ARIMA.	reduced by 22% vs Prophet.	integrated heterogeneous data; demonstrated scalability across Asia-Pacific destinations.	preprocessing pipeline; no decision-support dashboard for policymakers.
Web Search Data for Tourism Forecasting	Lee (2025)	Integrated behavioral data (Google search queries) with traditional models to improve East Asian tourism forecasts.	Monthly arrivals and Google Trends data (East Asia, 2015–2022).	SARIMAX with exogenous regressors (search indices). Seasonal dummies and lagged arrivals included. Benchmarked against SARIMA.	RMSE ≈ 11.2; MAPE ≈ 7.8%.	Proved predictive value of web search data; outperformed SARIMA baseline; improved short-horizon forecasts.	Limited to one country; SARIMAX lacks robustness under structural breaks; no visualization or DSS deployment.
ANN and RF Forecasting for Post-COVID Recovery	Kayral, Sarı, & Aktepe (2023)	Applied ML methods (ANN, RF) to predict arrivals and income post-COVID in Turkey.	Monthly arrivals and income statistics (Turkey, 2015–2022).	ANN with 3 hidden layers (ReLU activation, backpropagation, Adam optimizer). RF with 500 trees and bootstrap	ANN MAPE ≈ 8.3%; RF MAPE ≈ 9.5%; SARIMA MAPE ≈ 12.6%.	ANN reduced errors by 34% vs SARIMA; RF robust against outliers; showed ML superiority in	Limited to Turkey; ANN "black box" reduces interpretability; no deployment or cross-city comparability.

		Benchmarked against SARIMA baseline.		sampling. Features: lagged arrivals, seasonal dummies, exogenous income. Benchmarked via 10-fold CV.		volatile post-COVID recovery.	

Table 4 Similar Systems and their comparison in detail

In conclusion, research into tourism demand forecasting during and after the COVID-19 pandemic demonstrates the increasing reliance on machine learning (ML) and hybrid approaches to manage uncertainty in international arrivals and recovery patterns. To capture temporal dependencies and behavioral shifts in tourist flows, studies have effectively combined ensemble and deep learning techniques, including CNN-GRU architectures, LSTM, and tree-based models, with traditional econometric models (He et al., 2025; Nguyen et al., 2024). These systems demonstrate distinct improvements in responsiveness and accuracy, especially when external drivers like search engine indices, mobility data, and COVID-19 case counts are incorporated (Li et al., 2024; Dimitriadou & Gogas, 2025).

Significant limitations still exist in spite of these accomplishments. With models that frequently lack transferability across various tourism markets, many current systems are still geographically limited and concentrate on specific destinations (such as Taiwan, Indonesia, or Vietnam) (Afrianto & Wasesa, 2022; Moreno-Izquierdo et al., 2024). Furthermore, few systems offer interpretable outputs that directly support industry decision-making or short-horizon policy interventions, yet ensemble frameworks and hybrid models increase predictive accuracy (Bi et al., 2024; Zheng et al., 2025). The majority of systems are research prototypes rather than operational dashboards, which limits their practical deployment value for stakeholders tasked with managing tourism recovery in unstable conditions. This also limits technical scalability (Adekuajo et al., 2025).

These gaps highlight the practical need for forecasting systems that support real-time integration of heterogeneous data streams, strike a balance between interpretability and accuracy, and provide insights in an easily accessible format that aids in decision-making. In order to ensure usability and continuity, the Tourism Confidence Index (TCI) proposed in this study aims to do just that by standardizing metrics across several Asia Pacific cities, incorporating both statistical baselines (like Seasonal Naïve) and advanced machine learning models (like Ridge Regression, Random Forest, and XGBoost), and embedding the results within a lightweight, deployable dashboard (via Streamlit).

The importance of tourism recovery forecasting goes beyond its technical and operational contributions to include wider societal outcomes, especially in light of the Sustainable Development Goals (SDGs) of the UN. Resilient tourism directly supports SDG 3: Promoting safe travel, reviving tourism-dependent jobs, and lowering socioeconomic stressors that arose during extended shutdowns are all ways to promote good health and well-being (Velu et al., 2022; Sulong & Abdullah, 2023). In addition to helping to manage demand shocks, precise short-horizon forecasts enable policymakers to put in place health precautions and resource allocation plans that promote the general welfare.

The examination of comparable systems thus reveals advancements as well as enduring problems: improved accuracy without interpretability, methodological sophistication without broad generalizability, and research prototypes without scalable deployment. The TCI system, which attempts to combine accuracy, interpretability, and accessibility into a strong framework for monitoring and directing post-pandemic tourism recovery in Asia Pacific hubs, has a clear justification thanks to the gaps that have been identified.

2.4 Technical Research

2.4.1 Project Hardware and Software

A MacBook Air M2 with 16 GB of RAM running macOS Ventura is used to develop and carry out the project. This hardware platform was chosen because it strikes a balance between portability, computational efficiency, and energy sustainability—features that are essential for both practical deployment and scholarly research. High-performance parallel computation appropriate for machine learning workflows is provided by the Apple M2 chip, which combines a 10-core GPU and an 8-core CPU into a single architecture (Apple, 2023). This is particularly helpful when training models that need iterative optimization and efficient matrix operations, like Ridge Regression, Random Forest, and XGBoost.

With 16 GB of unified memory, the system can manage medium-to-large-scale datasets without relying too much on disk swapping, such as monthly visitor arrivals across several Asia Pacific hubs. Additionally, the NVMe-based SSD storage guarantees quick write and retrieve speeds, facilitating effective time-series data transformation and preprocessing. By enabling sustainable, mobile research practices with lower heat and energy demands, the M2 system-on-chip aligns with SDG 3 by offering optimized power efficiency in contrast to high-end GPUs used for deep learning workloads (Zhou et al., 2023).

All things considered, the MacBook Air M2 offers a dependable computing environment that strikes a balance between portability and performance. Its capabilities are adequate for developing interactive dashboards, performing ensemble models, and fine-tuning hyperparameters—all of which support the Tourism Confidence Index (TCI)—but it is not designed for large-scale deep learning clusters.

2.4.2 Programming Languages

For frontend interaction, the project mostly uses JavaScript (through Streamlit's React foundation) and Python for backend modelling. Thanks to its vast ecosystem of machine learning, forecasting, and visualization libraries, Python continues to be the most popular language for data science (Lau et al., 2022). Model training (Scikit-learn, XGBoost), evaluation (Statsmodels, Scipy), and preprocessing (using Pandas and NumPy) are all made easier by its ease of use and adaptability. Continuous library updates and reproducibility are provided by the Python open-source community, which is crucial for guaranteeing the project's longevity.

Despite being Python-based, Streamlit uses a frontend engine driven by JavaScript and React to create interactive, responsive dashboards without requiring manual UI programming. This

bridges the gap between technical outputs and policy decision-making by guaranteeing that outputs—like forecast plots, error metrics, and the Tourism Confidence Index—are communicated in a clear and understandable manner. Accessible forecast communication is essential from the standpoint of SDG 3, as it enables stakeholders to match recovery plans with population welfare and safety regulations pertaining to health (Velu et al., 2022).

With Python guaranteeing methodological rigor in model development and Streamlit's hybrid foundation offering accessibility and ease of deployment, the technical system thus directly supports tourism recovery management.

2.4.3 Integrated Development Environments (IDEs)

The project makes use of Visual Studio Code (VS Code) for dashboard integration and Jupyter Notebook for model development.

A literate programming environment that combines code execution with experimentation, storytelling, and visualization is provided by Jupyter Notebook. This makes it possible to transparently and reproducibly document exploratory data analysis (EDA), model iteration, and error metric evaluation. Because it promotes iterative hypothesis testing and makes collaboration easier, Jupyter's notebook format has gained widespread use in both academic and industrial data science (Rule et al., 2020).
For web-based deployment, Visual Studio Code is the main IDE. Backend Python scripts and Streamlit frontend elements can be seamlessly integrated thanks to its modular plugin system, Git integration, and debugging tools. The lightweight design of Visual Studio Code guarantees stability when building a dashboard while handling several interdependent elements, including visualization libraries, Streamlit widgets, and model imports.

The TCI project's dual emphasis of methodological transparency (research integrity) and operational accessibility (decision-support delivery) is reflected in the combination of Jupyter for research and Visual Studio Code for deployment. Since it guarantees that research outputs are not isolated in technical environments but are instead made actionable through public-facing systems, this duality aligns with SDG 3.

2.4.4 Tools and Libraries

A carefully selected collection of Python libraries and tools that are suited to the various stages of the CRISP-DM process—preparation, modelling, evaluation, and deployment—are used in this project.

Data preprocessing: Time-series arrivals data can be handled in an organized manner with Pandas and NumPy, which support dummy encoding, rolling averages, and lag creation. In order to capture seasonality and persistence in tourism demand, these computationally efficient operations are essential (McKinney, 2022).

Modelling: Scikit-learn offers Random Forest and Ridge Regression implementations, guaranteeing access to cross-validation and standardized hyperparameter optimization tools. Scalable gradient boosting algorithms tailored for tabular data are added by XGBoost; these algorithms are especially good at capturing nonlinear recovery dynamics in places like Hong Kong (Chen et al., 2021).
In order to ensure interpretability and baseline comparisons, Statsmodels offers a foundation for implementing Seasonal Naïve benchmarks.

Visualization: Arrivals, error distributions, and comparative model performance can be visualized in multiple layers using Matplotlib, Seaborn, and Plotly. By converting complicated data into insights that stakeholders can act upon, visual communication of error metrics is essential to transparency and supports SDG 3.

Deployment: Streamlit removes obstacles between research prototypes and practical tools by integrating backend Python models into interactive dashboards.

By ensuring scalability (through XGBoost), interpretability (through Seasonal Naïve), and accessibility (through Streamlit), this toolkit positions the TCI system as both technically sound and practically useful.

2.4.5 Operating System

The project is being developed on macOS Ventura, which was selected due to its compatibility with modern machine learning libraries, stability, and Unix-based foundation. Python virtual environments and package managers like Homebrew are natively supported on

macOS, making reproducibility and dependency management easier. Data confidentiality during preprocessing and analysis is ensured by its security infrastructure, which lowers malware risks and includes Gatekeeper and system integrity protection (Apple, 2023).

By promoting low-barrier research productivity and sustainable computing practices, the operating system's lightweight design also extends battery life, allowing mobile research in a variety of settings. This indirectly supports SDG 3.

2.4.6 Environment for Web Server and Deployment

The Tourism Confidence Index dashboard is deployed by the project using Streamlit's built-in lightweight web server. Streamlit abstracts backend complexity while guaranteeing real-time responsiveness, in contrast to more complex frameworks that need external servers. This is especially useful for interactively presenting stakeholders with updated tourism forecasts and error metrics.

Iterative testing is made possible by Streamlit's local Chrome run capability, and scalability is guaranteed by its cloud-based deployment compatibility if it is expanded for broader institutional use. Streamlit transforms static research into real-time decision-support tools by integrating sophisticated machine learning models into an intuitive user interface, bridging the gap between academic rigor and policy utility.

2.4.7 Web Browser

The main testing browser was chosen to be Google Chrome. With its strong support for JavaScript engines and contemporary rendering frameworks, Chrome continues to be the most popular browser worldwide (StatCounter, 2023). Time-series plots, forecast comparison panels, and TCI metrics are all smoothly visualized thanks to its compatibility with Streamlit. Additionally, the browser's developer tools offer crucial debugging features that optimize dashboard responsiveness on a variety of screen sizes.

The project promotes inclusivity and usability by guaranteeing accessibility through widely used browsers, which directly connect to SDG 3 by enabling recovery insights to be accessed by a variety of user groups without the need for specialized software.

2.4.8 Justification and Alignment with SDG 3

Every technical decision in this project was chosen for its ability to support a sustainable tourism recovery as well as its computational efficiency. Jupyter and VS Code promote transparency; lightweight deployment options lower barriers to stakeholder adoption; Python and Streamlit offer accessible, community-driven ecosystems; and the MacBook Air M2 guarantees portable yet potent research. Together, they support instruments that guide policies meant to hasten recovery, lower uncertainty, and protect livelihoods and well-being in the Asia Pacific tourism industry, all of which are in line with SDG 3: Good Health and Well-Being.

CHAPTER 3: METHODOLOGY

3.1 Introduction

This chapter outlines the approach used to create the Tourism Confidence Index (TCI) system, which examines the post-pandemic tourism recovery in three Asia Pacific cities: Hong Kong, Bangkok (a stand-in for Thailand), and Singapore. The foundation of a project is its methodology, which guarantees that every phase—from conception to deployment—is carried out methodically, reproducibly, and in accordance with the project's main goal.

The first section of the chapter discusses well-known data science approaches that have influenced the creation of forecasting and decision-support systems, including the Knowledge Discovery in Databases (KDD) methodology, the Sample, Explore, Modify, Model, Assess (SEMMA) process, and the Cross-Industry Standard Process for Data Mining (CRISP-DM). Each method is thoroughly explained, emphasizing its theoretical foundations as well as its real-world applications in forecasting tourism demand. The methods are then compared using technical, practical, and business-related criteria. As a result, CRISP-DM is chosen as the project's guiding framework.

CRISP-DM is directly applied to the TCI project in the second half of the chapter, which maps each of its six phases—business understanding, data understanding, data preparation, modelling, evaluation, and deployment—to the particular datasets, tools, and algorithms used. The conversation illustrates how CRISP-DM is the best option for predicting uneven tourism recovery trajectories because it guarantees systematic progression while retaining flexibility for refinement.

3.2 Methodology

For data-driven projects to retain coherence, validity, and practical relevance, methodology is crucial. The stakes are especially high when it comes to tourism forecasting because forecasting accuracy has a direct impact on business, government, and public health decision-making. Demand patterns change as a result of health restrictions, Traveller confidence, and macroeconomic factors, adding complexity to the post-pandemic recovery. In this situation, methodology serves as a safeguard against bias, poor generalization, and misinterpretation in addition to being a technical workflow.

To organize data science projects, a number of approaches have been developed. Among these, CRISP-DM, SEMMA, and KDD are well known for offering scalable, repeatable

frameworks. Their focus, scope, and suitability for real-world integration vary, despite the fact that they both aim to extract insights from data. As a result, each methodology is covered in detail below, with specific technical and practical implications related to the TCI project.

3.2.1 Introduction of Methodologies

CRISP-DM, or the Cross-Industry Standard Process for Data Mining

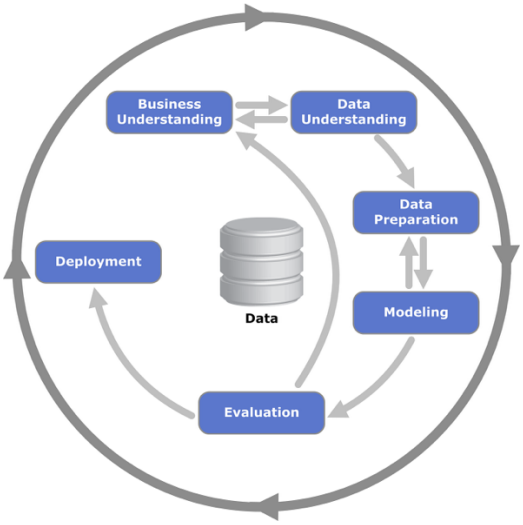


Figure 3 CRISP-DM Workflow (Shaun Chumbar,2024)

It is the most popular approach for decision-support systems and predictive modelling is CRISP-DM (Schröer et al., 2021). Business understanding, data understanding, data preparation, modelling, evaluation, and deployment are the six iterative phases into which it divides projects.

Theoretically, CRISP-DM's iterative loops are its strongest feature. For example, evaluation-related insights could lead to modifications in modelling or data preparation, allowing for ongoing improvement. This is especially helpful in unstable areas where recovery indicators change over time, like post-pandemic tourism.

Practically speaking, CRISP-DM corresponds exactly to the workflow of the TCI system:

- Business knowledge supports the definition of tourism recovery as an indicator of both well-being and the economy, which is in line with SDG 3.

- Understanding data relates to gathering Google Trends indices, hotel occupancy rates, and monthly visitor arrivals (2015–2024).

- Jupyter-coded feature engineering steps (lags, rolling averages, month dummies, and log transformations) reflect data preparation.

- Ridge Regression, Random Forest, XGBoost, and Seasonal Naïve are all included in the modelling process and are selected to test varying degrees of complexity.

- When series variance is low (as in Singapore in 2024), there is a clear rationale for prioritizing error metrics, and evaluation employs MAPE, MAE, RMSE, and R^2.

- The Streamlit dashboard is where deployment takes place, converting technical outputs into insights that stakeholders can use.

As a result, CRISP-DM offers the TCI project not only a theoretical framework but also a framework that can be immediately operationalized, guaranteeing that each stage can be tracked from raw data to an interactive decision-support tool.

SEMMA stands for Sample, Explore, Modify, Model, and Assess.

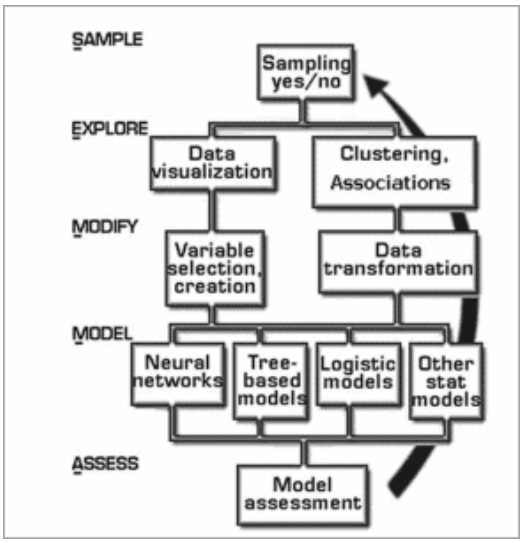


Figure 4 SEMMA Workflow Shaun Chumbar,2024)

The SAS Institute created the five-step framework known as SEMMA, which stands for sample, explore, modify, model, and assess (Padilla-Vento & Soria, 2025). According to theory, SEMMA is appropriate for projects that aim to find patterns in structured data

because it places a strong emphasis on modelling and exploratory tasks. Its scant attention to deployment and business objectives, however, has drawn criticism (Leidner & Reiche, 2023).

When applied practically to this project, SEMMA would look like this:

- For quicker experimentation, sampling might entail selecting subsets of arrivals and occupancy data (2015–2024), but this runs the risk of underutilizing the entire temporal range.
- Descriptive arrival visualizations and Google Trends indices to show recovery dips and rebounds would be examples of exploring.
- To better capture temporal dependencies, modifications would include the creation of month dummies, rolling means, and lag features (1, 3, 6, 12).

- These engineered features would be modeled using XGBoost, Random Forest, and Ridge Regression.
- The evaluation strategy already coded in Jupyter would be directly reflected in the assessment, which would rely on MAPE and RMSE.

The technical workflow utilized in this project could be largely replicated by SEMMA, but it does not carry over into deployment. Since a Streamlit dashboard is one of the project's deliverables, SEMMA is less appropriate as a guiding methodology because it lacks a deployment framework.

Knowledge Discovery Database (KDD)

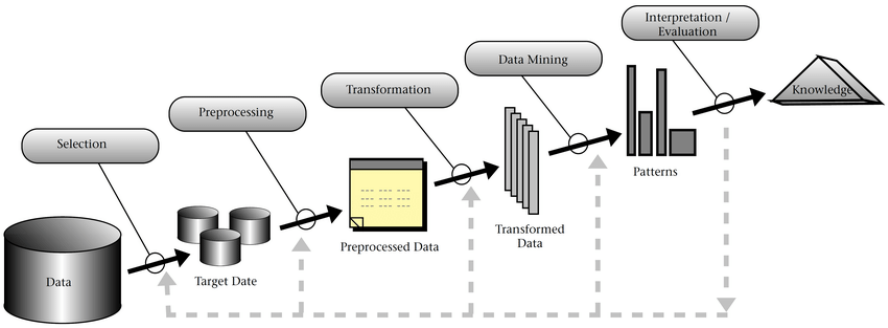


Figure 5 KDD Workflow Fayyad et al. (1996a))

A systematic framework for converting unstructured data into insightful and useful knowledge is offered by the Knowledge Discovery in Databases (KDD) methodology. It consists of a number of interrelated phases, each intended to guarantee methodical advancement from data collection to interpretation. Data selection is the first step in the process, during which pertinent datasets are located and collected from various sources (Daza et al., 2022). To guarantee dependability for additional analysis, the preprocessing step then tackles problems with data quality, such as missing values, noise, and inconsistencies (Vasiliev & Goryachev, 2022).

The data goes through a transformation stage after it has been cleaned, where it is reformatted into a format that is appropriate for mining; this frequently entails feature construction, dimensionality reduction, or normalization to bring out underlying patterns (Palacios et al., 2021). Advanced algorithms like classification, clustering, regression, or association rule learning are used to extract patterns and predictive models during the data mining stage (Chen et al., 2021). Ultimately, by comparing the mined patterns to real-world goals and presenting them in an intelligible manner, the interpretation and evaluation stage guarantees that they are significant, legitimate, and pertinent (Ortega-Guzmán et al., 2024). These phases work together to create a flexible and iterative process for finding insights in a variety of industries, from engineering and business intelligence to tourism and healthcare (Hall, 2022).

Through iterative discovery, KDD aims to convert unstructured data into meaningful knowledge (Ortega-Guzmán & Gutiérrez-Preciado, 2024). Data selection, preprocessing, transformation, mining, and interpretation are some of its phases. Theoretically, KDD works well for projects that try to find hidden patterns in big, diverse datasets (Butka et al., 2020). Its iterative process aligns with exploratory analysis in forecasting demand for tourism.

Based on a real-world mapping to this project, KDD would include:

- Deciding on search patterns, hotel occupancy, and arrivals as the main indicators.
- Addressing missing values during preprocessing (such as interpolating gaps in Hong Kong's data).
- Converting features into formats that can be used, like aggregated rolling means or log-transformed arrivals.
- Recovery dynamics through mining with machine learning models (Ridge, Random Forest, XGBoost).

- Interpreting the results as the recovery confidence trajectories of different cities.

3.2.2 Methodology Choice and Justification

The three methodologies are compared below, with justification for their suitability to the Tourism Insights project.

Characteristic	CRISP-DM	SEMMA	KDD	Relevance to TCI Project
End-to-End Coverage	Business understanding → deployment	Modelling-focused	Knowledge discovery	CRISP-DM ensures full workflow from objectives to dashboard
Iteration & Flexibility	High	Low	Moderate	Recovery forecasting requires iterative recalibration
Business Objective Alignment	Strong	Minimal	Limited	CRISP-DM connects to SDG 3 (well-being, resilience)
Data Handling	Robust preprocessing	Sampling-focused	Transformation emphasis	CRISP-DM handles lags, rolls, and seasonal dummies effectively
Deployment Guidance	Explicit	None	Weak	Dashboard deployment requires deployment phase
Scalability	High	Limited	Moderate	CRISP-DM scales across multiple cities consistently
Stakeholder Involvement	Strong	Weak	Weak	CRISP-DM ensures outputs are interpretable and actionable
Real-World Integration	Strong	Weak	Limited	CRISP-DM bridges research and decision-support

This assessment leads to the selection of CRISP-DM as the project's methodology. There are three reasons for this:

- **Technical Fit**: Jupyter's workflow, which used Ridge, Random Forest, XGBoost, and Seasonal Naïve to engineer and test features, is aligned with CRISP-DM's explicit focus on data preparation, modelling, and evaluation.
- **Practical Relevance:** CRISP-DM guarantees that the project ends with an interactive, deployable system via Streamlit, not merely a research prototype, in contrast to SEMMA or KDD.
- **SDG 3 alignment:** By lowering uncertainty for Travellers, companies, and policymakers, CRISP-DM's business understanding and deployment phases guarantee that outputs are linked to actual recovery, enhancing well-being.

In order to maintain the project's technical rigor, practical deployability, and social relevance, CRISP-DM was selected as the guiding methodology.
A number of considerations support the selection of CRISP-DM:

- Domain Independence: CRISP-DM is especially well-suited to the multidisciplinary nature of tourism recovery studies because it can be applied to a wide range of industries and research fields (Wirth & Hipp, 2000).
- Iterative Process: The cyclic and feedback-driven structure of CRISP-DM is in line with the dynamic nature of tourism recovery analysis, which calls for iterative model refinement based on incoming data.
- Emphasis on Business Understanding: According to Mariscal, Marbán, and Fernández (2010), CRISP-DM places a high priority on establishing business objectives and converting them into data mining goals. This is essential to guaranteeing that the research outputs are in line with actual tourism recovery needs.
- Strong Evaluation Phase: The integrated evaluation phase guarantees that models are evaluated for both business relevance and predictive power, which is a crucial prerequisite for significant results in the post-pandemic tourism industry.

These factors led to the conclusion that CRISP-DM was better than other approaches like KDD (Knowledge Discovery in Databases) and SEMMA (Sample, Explore, Modify, Model,

Assess). These approaches are powerful, but they are either too general or too tool-specific (for example, SEMMA's association with SAS software) for the open-source and academic nature of this project (Piatetsky-Shapiro, 1996; SAS Institute, 2001).

3.2.3 CRISP-DM Methodology

This project was guided by the Cross-Industry Standard Process for Data Mining (CRISP-DM) framework because of its structured, iterative, and generally accepted suitability for data-driven forecasting tasks (Schröer et al., 2021). Business understanding, data understanding, data preparation, modelling, evaluation, and deployment are the six stages that make up CRISP-DM. To guarantee methodological rigor, alignment with project objectives, and flexibility in the face of post-pandemic tourism recovery uncertainties, each phase was applied methodically.

Business Understanding

The basis for matching data-driven approaches with more general organizational or societal goals is provided by the business understanding stage of CRISP-DM (Schröer et al., 2021). Developing a smart traveller insights system that could predict post-pandemic recovery in Bangkok, Singapore, and Hong Kong was the main goal of this project. This project aligns with the United Nations Sustainable Development Goal 3 (Good Health & Well-Being) by positioning recovery as a determinant of public well-being and a driver of economic resilience, in contrast to traditional economic models of tourism. For populations that rely on tourism, prior studies have shown that steady tourism flows enhance traveller confidence, lower uncertainty, and promote mental health (Hall et al., 2021; Vanneste, 2022).

This approach to recovery forecasting pushes the project beyond financial analysis and adds a well-being dimension that was not taken into account in previous forecasting systems.
The business problem was stated as follows: How can the recovery of international tourism be predicted with enough short-term precision and interpretability to serve as a tool for decision-making in various city contexts? This query aligns with earlier studies on tourism that highlight the value of short-horizon projections in handling shocks and emergencies (Song et al., 2021). Unprecedented disruption was caused by COVID-19, which created volatility that necessitated both proactive forecasting tools that could direct recovery strategies and retrospective analysis.

Early on, limitations were recognized. Kuala Lumpur, for example, was first included but later removed after data quality evaluations revealed that missing months and inconsistent reporting jeopardized reliability. These choices are in line with CRISP-DM best practices, which stress that viability needs to be based on data realities (Casonatto et al., 2024). Similar to this, a technical limitation of Singapore's low-variance 2024 test set led to the preference for error-based evaluation metrics (MAPE, MAE) over variance-sensitive ones like R2 (Shang et al., 2021).

This phase's goals were as follows:

- Employing a variety of statistical and machine learning techniques to predict monthly arrivals uniformly across cities.
- Comparing models with different recovery paths.
- Results are released using a dashboard based on Streamlit, guaranteeing that they are useful tools for decision-making rather than solitary research products (Truong et al., 2022).

Thus, both technical and practical definitions of the success criteria were made. According to industry standards for demand forecasting, it was technically acceptable to achieve forecasting errors below 10% MAPE (Li et al., 2022). In practice, success was determined by how easily the system could be interpreted and accessed, guaranteeing that the results could guide recovery planning in actual tourism governance. The business understanding phase verified that the project was not just a computational exercise but also a strategically aligned intervention to support resilience and well-being by placing technical decisions within larger frameworks for economic recovery and public health.

Data Understanding

In the data understanding phase, datasets were gathered, described, and examined to make sure they were appropriate for forecasting. Three primary data streams were included:

- Data on Tourist Arrivals
  Bangkok, Singapore, and Hong Kong's monthly foreign arrivals (2015–2024). acted as the dependent variable, recording patterns of recovery, shocks, and seasonality.

- Rates of Hotel Occupancy
  In order to validate arrivals data and improve recovery signal detection, monthly hotel occupancy data offered an additional proxy for tourism demand (Wang et al., 2023).

- Trends Indicators on Google
  High-frequency, real-time proxies for travel intent were provided by search activity data (e.g., "Bangkok," "Singapore," and "Hong Kong" under the category "Travel"). When official data did not keep pace with behavioral changes during pandemic disruptions, these were especially helpful (Sato et al., 2022).

To find patterns, anomalies, and structural breaks, exploratory data analysis (EDA) used plots, descriptive statistics, and correlation analysis. Among the difficulties were sudden drops during border closures, a slower rate of recovery in Hong Kong than in Bangkok and Singapore, and occasionally insufficient reporting. Cross-validation across sources, outlier detection, and missing value analysis were used to evaluate the quality of the data.

In line with recent calls for blended data streams to improve resilience under crisis conditions, the combination of traditional indicators (arrivals, occupancy) and non-traditional indicators (search trends) reflects methodological innovation in tourism forecasting (Song & Li, 2021).

Data Preparation

The most intensive step in CRISP-DM is data preparation, which entails converting unprocessed data into appropriate modelling inputs (Saltz, 2021). The data preparation for this project took into account the seasonality, variance disparities, and structural shocks that are present in tourism recovery datasets. Raw data was converted into a structured format appropriate for modelling during the data preparation stage.

Data Cleaning

- Where necessary, values for missing arrivals were interpolated.
- True shocks (such as border closures) were kept, but outliers brought on by reporting errors were fixed.
- Data from Google Trends and hotel occupancy were normalized to correspond with monthly frequency.

Feature Engineering

- Lag Features: To account for autoregressive effects, past arrivals at 1, 3, 6, and 12 months are used.
- Rolling Means: Averages of three and six months to reduce volatility in the short term.
- Seasonal Dummies: Indicators unique to a given month that capture seasonality.
- Log transformation: enhanced regression performance and stabilized variance.
- Cross-Domain Features: Exogenous predictors such as hotel occupancy and Google Trends are included, demonstrating that heterogeneous features improve forecasting accuracy (Chen et al., 2022).

Modelling

Creating predictive models appropriate for encapsulating the nonlinear and diverse dynamics of post-pandemic tourism was the task of the modelling phase. Four strategies were used:

Ridge Regression: Selected due to its interpretability and resilience to multicollinearity, especially when lagged variables and seasonal dummies are included. High-dimensional feature spaces were stable thanks to regularization (Hastie et al., 2021).
Random Forest (RF): Chosen for its capacity to aggregate decision trees and model nonlinear interactions and recovery heterogeneity. In determining predictor relevance, its feature importance outputs also offered interpretability (Zhou et al., 2022).

Extreme Gradient Boosting (XGBoost): Integrated as a cutting-edge ensemble technique that excels at nonlinear forecasting. Its iterative boosting mechanism improved accuracy for volatile recovery trajectories by minimizing residual errors (Chen et al., 2021).
Seasonal Naïve (Baseline): Used as a standard, taking advantage of the high seasonality in demand for travel. By placing advanced model performance against a straightforward criterion, this guaranteed methodological rigor (Song & Li, 2021).

Training Design

- Data Split: Validation (2023), Test (2024), and Training (2015–2022).

- Leak prevention and time-aware evaluation were guaranteed by walk-forward validation.
- Validation error, mainly MAPE and RMSE, was used to optimize hyperparameters (such as penalty terms in Ridge, number of trees in RF, and learning rate in XGBoost).

Transparency, robustness, and methodological completeness were all balanced in the modelling portfolio by combining interpretable statistical techniques with sophisticated machine learning ensembles and a baseline comparator.

Evaluation

The CRISP-DM evaluation phase makes sure that created models are thoroughly examined in relation to project goals and verified for accuracy and usefulness. In line with current best practices in tourism forecasting, this project's evaluation was organized around a mix of error-based and variance-based metrics (Assaf et al., 2022; Li et al., 2022).

The metrics listed below were chosen:

- The primary evaluation metric, Mean Absolute Percentage Error (MAPE), was chosen because it is easy to understand as a percentage and is frequently used in forecasting tourism demand. For short-horizon forecasts, a target of less than 10% MAPE was established as an acceptable benchmark (Chen et al., 2022).
- Mean Absolute Error (MAE): Added scale-sensitive error magnitudes to MAPE, which is especially helpful in comprehending actual arrival deviations.
- Root Mean Squared Error (RMSE): Added to ensure that extreme deviations were taken into account by penalizing larger errors more severely.
- In low-variance test sets, the coefficient of determination (R2), which is reported for completeness, should be interpreted cautiously as it may skew performance evaluation (Shang et al., 2021).

The combination of metrics addressed methodological issues brought up in recent literature on crisis-sensitive tourism forecasting by enabling a balanced evaluation of both relative accuracy (MAPE, MAE) and variance-explained (R^2) (Hall, 2022).

Evaluation Design

The evaluation was conducted using a time-conscious validation framework:

- Training period: pre-pandemic, pandemic onset, and recovery initiation, 2015–2022.
- 2023 is the validation set (used for intermediate evaluation and hyperparameter tuning).
- 2024 (an unspecified future time frame that represents short-horizon deployment conditions) is the test set.

Instead of overfitting historical data, this structure made sure that models were assessed based on their capacity to generalize to recovery conditions. In accordance with methodological recommendations for time series forecasting, walk-forward validation was used to stop leakage across temporal boundaries (Song & Li, 2021).

Alignment to Project Goals

Two success levels served as the framework for evaluation:

- Technical success: Forecast errors (MAPE < 10%) are consistently low across cities and models.
- Making sure the results were comprehensible and appropriate for incorporation into the Tourism Confidence Index (TCI) dashboard was a practical success.

The evaluation strategy recognized that the value of forecasts goes beyond numerical accuracy by integrating quantitative metrics with qualitative interpretability assessment. This dual focus helped to address the resilience and well-being goals highlighted in SDG 3 by ensuring alignment with the project's overarching goal of lowering uncertainty in the tourism recovery and fostering stakeholder confidence (Vanneste, 2022).

Deployment

The models are operationalized and the forecasting outputs are made available for decision-making during the deployment phase. The deployment of this project focused on using Streamlit to create an interactive decision-support dashboard that allowed forecasts, diagnostics, and evaluation metrics to be displayed in an understandable and user-friendly manner.

System Integration

The Tourism Confidence Index (TCI) dashboard was updated to incorporate the results of the

Seasonal Naïve, Random Forest, XGBoost, and Ridge Regression models. Forecasts and diagnostic tools like error summaries and seasonal plots could be seen on the dashboard's user interface. Streamlit is a good environment for connecting technical modelling and real-world application since it enables quick prototyping and real-time updates without requiring complicated web infrastructure (Akter et al., 2023).

Technical Justification

The deployment pipeline was created with accessibility and efficiency in mind:

- **MacBook Air M2 (16 GB RAM, Ventura OS):** Served as the deployment and development environment which had enough space for dashboard integration and model training. Its specifications were sufficient for the regression- and tree-based models used in this study, despite not being optimized for deep learning.
- **Streamlit as a Web Server:** chosen over alternatives like Dash or Flask because of its easy integration with Python workflows, lightweight design, and support for iterative deployment and testing (Truong et al., 2022).
- **Chrome Web Browser:** chosen for testing and demonstration, guaranteeing device compatibility and dependable result presentation.

Maintenance and Scalability

Streamlit facilitates scalability by integrating with cloud environments (such as AWS and GCP), even though the system was only implemented as a prototype. This guarantees the possibility of the TCI dashboard being hosted as a live tool in the future, with data being updated and models being retrained on a regular basis. The emphasis on continuous monitoring and iterative improvement found in methodological best practices for decision-support systems aligns with this scalability (Zhang et al., 2023).

The idea of converting scholarly outputs into practical tools is also reflected in deployment from a societal standpoint. The deployment guarantees that uncertainty is decreased for businesses, Travellers, and policymakers by offering an easily navigable interface for interacting with forecasts. Because less uncertainty promotes resilience and psychological well-being in communities that depend on tourism, this directly advances SDG 3 (Hall, 2022).

Summary

The process for creating a Smart Traveller Insights System to predict the post-pandemic tourism recovery was described in this chapter. The chapter explained the methodical procedures used, starting with defining the business context and continuing through data preparation, model building, performance evaluation, and result deployment, using CRISP-DM as the overarching framework.

The approach placed a strong emphasis on:

- **Structured business alignment,** where forecasting was presented as a socially beneficial activity in addition to an economic one.
  thorough data preparation that captures recovery dynamics and temporal dependencies using log transformations, rolling averages, seasonal dummies, and lag features.

- **Several modelling techniques,** including baseline methods (Seasonal Naïve), machine learning techniques (Random Forest, XGBoost), and statistical techniques (Ridge), guarantee robustness across various city trajectories.

- **Thorough assessment framework,** with time-aware partitioning used to represent real-world circumstances and error-based metrics given priority for accuracy assessment.

- **Accessible deployment,** in which forecasts were operationalized in an interactive and comprehensible manner using the Tourism Confidence Index dashboard based on Streamlit.

This strategy allowed the project to bridge the gap between the practical decision-support utility and the technical forecasting rigor. The methodology's emphasis on interpretability, accessibility, and contextual relevance supported the resilience of tourism ecosystems while also guaranteeing accurate short-horizon forecasting. This project's contribution to SDG 3 is strengthened by its dual technical and societal orientation, which emphasizes the importance

of data-driven tools in fostering confidence, lowering uncertainty, and improving public well-being during the post-pandemic recovery.

CHAPTER 4: DESIGN AND IMPLEMENTATOIN

4.1 Introduction

The project's conceptual and methodological underpinnings are translated into a tangible data-driven forecasting pipeline during the design and implementation phase. This chapter details the Smart Traveller Insights System's real-world implementation, from gathering datasets to building models. The three Asia-Pacific locations that were chosen—Singapore, Hong Kong, and Bangkok (a stand-in for Thailand)—are described along with the methods for data collection, preprocessing, exploratory data analysis, and model development.

The exploratory data analysis carried out to comprehend the statistical characteristics and dynamics of the datasets is also presented in this chapter. The recovery patterns in the three cities are revealed by visualizations like correlation heatmaps, seasonal decomposition, and temporal trends. Instead of being viewed as black-box processes, these analyses guarantee that the ensuing models are based on empirical observations.
Google Trends data and official international arrivals statistics were the two main dataset types gathered for this project's chosen Asia-Pacific cities.

The chapter concludes with a description of how various forecasting models are implemented. While machine learning models like Ridge Regression, Random Forest, and XGBoost were used to capture nonlinearities and complex relationships in the data, traditional baselines like the Seasonal Naïve approach were included for benchmarking purposes (Kumar, Misra, & Chan, 2022; Gunter & Önder, 2023). Prophet's extensive use in irregular tourism time series led to its inclusion as a structural benchmark. These models work together to create the Tourism Confidence Index (TCI) dashboard and serve as the foundation of the forecasting system.
In conclusion, by describing the technical implementation of data collection, preprocessing, comprehension, and modelling, this chapter closes the gap between methodology and application. It provides a clear explanation of how unprocessed data is converted into useful insights, guaranteeing academic rigor and reproducibility.

4.2 Data Collection

A panel of monthly tourism-related datasets for Singapore, Hong Kong, and Bangkok (a proxy for Thailand) spanning January 2015 to December 2024 is used in this project. Reliability and reproducibility were guaranteed by the datasets' acquisition from Google Trends, statistical organizations, and official tourism boards. Three types of variables were gathered:

- Visitor Arrivals (target variable): National tourism authorities' monthly counts of foreign visitors.

- Monthly hotel occupancy percentages derived from official hotel and tourism statistics that represent demand for lodging are known as supply-side proxy hotel occupancy rates.

- The demand-side proxy, Google Trends Indices: Indicators of search interest for tourism-related keywords (such as "flights to Singapore," "hotels in Bangkok") that capture potential travel intent.

City	Coverage Period	Rows	Core Variable (Target)	Auxiliary Variables	Missing %	Source(s)
Singapore	2015–2024	120	Visitor Arrivals	Hotel Occupancy Rate; Google Trends	<1%	Singapore Tourism Board (2024); Google (2024) https://stan.stb.gov.sg/content/stan/en/tourism-statistics.html https://trends.google.com/trends/
Hong Kong	2015–2024	120	Visitor Arrivals	Hotel Occupancy Rate; Google Trends	~2%	Hong Kong Tourism Board (2024); Census and Statistics Department, HK (2024); Google (2024) https://partnernet.hktb.com/sea/en/home/index.html https://www.censtatd.gov.hk/en/ https://trends.google.com/trends/

Bangkok (Thailand proxy)	2015–2024	120	Visitor Arrivals	Hotel Occupancy Rate; Google Trends	~3%	Bank of Thailand (2024); Google (2024) https://www.bot.or.th/ https://trends.google.com/trends/

Table 5 Data collection with sources

4.3 Initial Data Understanding

4.3.1 Variables

Five main variables covering the years 2015–2024 make up the Bangkok dataset, which serves as the model for data comprehension. The attributes, their data types, completeness, and designated purpose within the modelling framework are shown in Table 5.

Column	Data Type	Non-Null Count	Missing %	Example Value	Purpose
year	int64	120	0.0	2015	Date/Temporal
month	int64	120	0.0	1	Date/Temporal
visitor_arrivals	float64	120	0.0	2,613,700	Target (Visitor Arrivals)
hotel_occupancy	float64	120	0.0	72.7	Auxiliary (Hotel Occupancy)
google_trends	int64	120	0.0	89	Auxiliary (Google Trends Index)

Table 6 Variables of the bangkok dataset

The target variable for the forecasting task is the visitor_arrivals column, which shows the monthly volume of foreign visitor arrivals. This series serves as the dependent component of the predictive models and measures the size of the demand for inbound tourism.

As a percentage, the hotel_occupancy variable serves as an auxiliary supply-side indicator. It provides information about actual demand conditions and shows how accommodation capacity is being used. This is a significant explanatory factor because higher occupancy rates

are generally associated with higher arrivals.

An additional demand-side proxy, the google_trends variable is derived from aggregated search interest indices for keywords related to tourism. Online search activity frequently predicts actual travel behavior, as mentioned in recent literature on tourism analytics, thus offering forward-looking indicators of demand for travel.

As temporal identifiers, the year and month variables make it easier to construct seasonal and cyclical features (such as lag structures and monthly dummies). These are crucial for feature engineering and identifying seasonality in visitor flows, but they are not predictive in and of themselves.

The dataset's 0.0% missingness rate across all columns shows that every variable is complete and has no observed missing values. When combined, these variables offer a logical basis for descriptive analysis as well as the creation of predictive features during the preprocessing phase that follows.

4.3.2 Exploratory Data Analysis (Bangkok Exemplar)

STEP 1: Importing libraries

Using Jupyter, we import the libraries necessary.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import os
```

Figure 6 Import the libraries

STEP 2: Load Dataset

By utilizing the path name, we get access to the dataset on jupyter.

```
# Load dataset
df = pd.read_csv("/Users/aman/Downloads/bangkok_2015_2024_final.csv", parse_dates=True, index_col="date")
df
```

Figure 7 loading dataset

These are the first three rows of the Bangkok dataset.

date					
2015-01-01	2015	1	2613700.0	72.70	89
2015-02-01	2015	2	2664220.0	71.99	84
2015-03-01	2015	3	2555360.0	68.31	80

Figure 8 First three rows of data

STEP 3: Univariate Analysis

The univariate exploratory data analysis performed on the Bangkok dataset is shown in the code snippet that follows. In order to investigate the marginal distributions of the three variables—visitor arrivals, hotel occupancy rates, and Google Trends indices—the analysis first creates a monthly datetime index using the year and month columns. Each variable's frequency distribution was examined using histograms, and a temporal overview was given by time-series plots. The mean visitor arrivals were totaled by calendar month in order to account for seasonality.

```
# === UNIVARIATE ANALYSIS ===

# 1) Histogram of arrivals
plt.figure()
bk["visitor_arrivals"].dropna().hist(bins=20)
plt.title("Bangkok — Visitor Arrivals Distribution")
plt.xlabel("visitor_arrivals"); plt.ylabel("Frequency")
plt.tight_layout(); plt.show()
plt.savefig(os.path.join(OUTDIR, "BKK_hist_arrivals.png")); plt.close()

# 2) Time-series plot of arrivals
plt.figure()
bk["visitor_arrivals"].plot()
plt.title("Bangkok — Visitor Arrivals Over Time (2015–2024)")
plt.xlabel("Date"); plt.ylabel("visitor_arrivals")
plt.tight_layout(); plt.show()
plt.savefig(os.path.join(OUTDIR, "BKK_ts_arrivals.png")); plt.close()

# 3) Seasonality — average arrivals by month
plt.figure()
bk.groupby(bk.index.month)["visitor_arrivals"].mean().plot(kind="bar")
plt.title("Bangkok — Average Visitor Arrivals by Month")
plt.xlabel("Month (1–12)"); plt.ylabel("Average visitor_arrivals")
plt.tight_layout(); plt.show()
plt.savefig(os.path.join(OUTDIR, "BKK_seasonality_arrivals.png")); plt.close()
```

Figure 9 Univariate analysis code

```
# Time series plots

bk["hotel_occupancy"].plot(title="Hotel Occupancy Over Time")
```

Figure 10 Univariate analysis for hotel occupancy

```
bk["google_trends"].plot(title="Google Trends Over Time")
```

Figure 11 Univariate analysis code for google trends

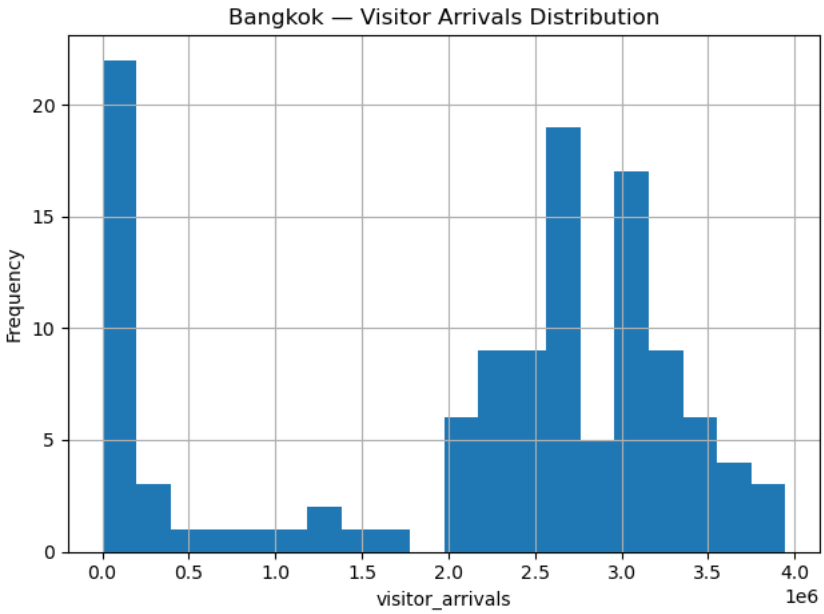


Figure 12 Bangkok Arrivals histogram

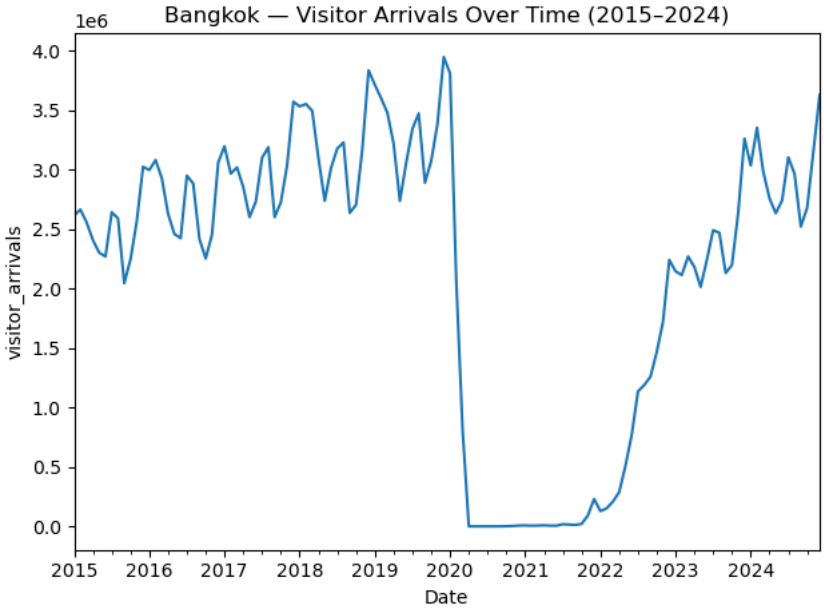


Figure 13 Bangkok Arrivals over time

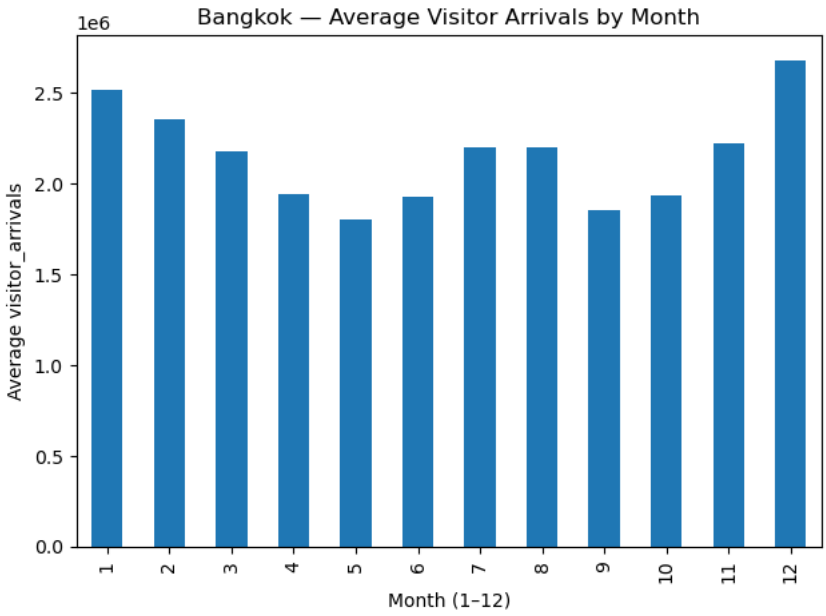


Figure 14 Average Visitors Arrivals for bangkok

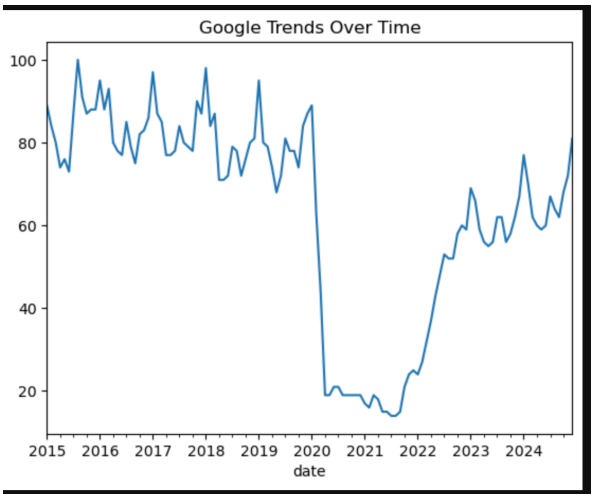


Figure 15 Google trends series plot over time

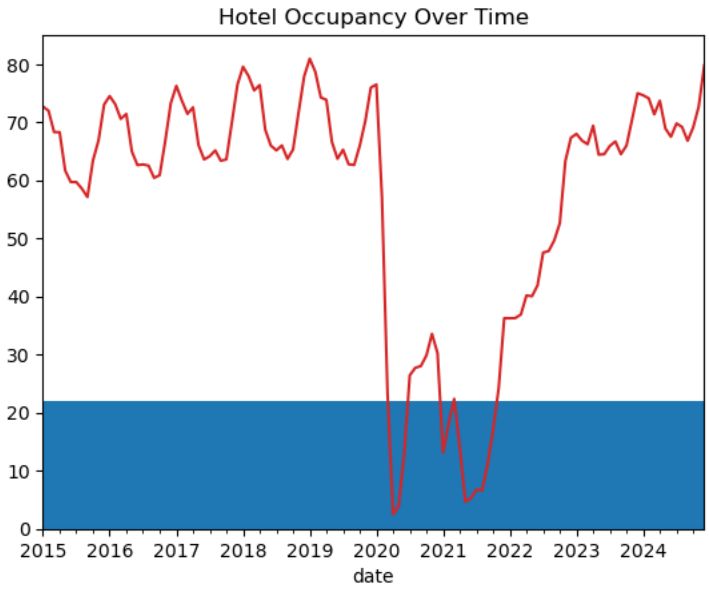


Figure 16 Hotel occupancy series plot over time

Visitor Arrivals: The histogram displays a distribution that is skewed to the right, with a few extreme peak values and the majority of monthly arrivals grouped below the long-term mean. Strong seasonality and a dramatic structural collapse in 2020 are evident in the time-series which is in line with travel restrictions brought on by the pandemic (Qiu et al., 2021).

Hotel Occupancy: shows a roughly bell-shaped distribution with some dispersion at lower levels and a centre of 60–70%. The time-series plot shows how the demand for lodging decreased in 2020 before gradually increasing again in 2022.

Google Trends: Significant drops in internet search activity during the pandemic are depicted in the time-series suggesting that demand-side interest was extremely sensitive to travel restrictions.

STEP 4: Bivariate Analysis

```
[32]: # Scatter: Arrivals vs Hotel Occupancy
      plt.scatter(bk["hotel_occupancy"], bk["visitor_arrivals"])
      plt.title("Visitor Arrivals vs Hotel Occupancy")

      # Scatter: Arrivals vs Google Trends
      plt.scatter(bk["google_trends"], bk["visitor_arrivals"])
      plt.title("Visitor Arrivals vs Google Trends")

      # Correlation matrix
      corr = bk[["visitor_arrivals", "hotel_occupancy", "google_trends"]].corr()
      print(corr)
```

Figure 17 Bivariate analysis code

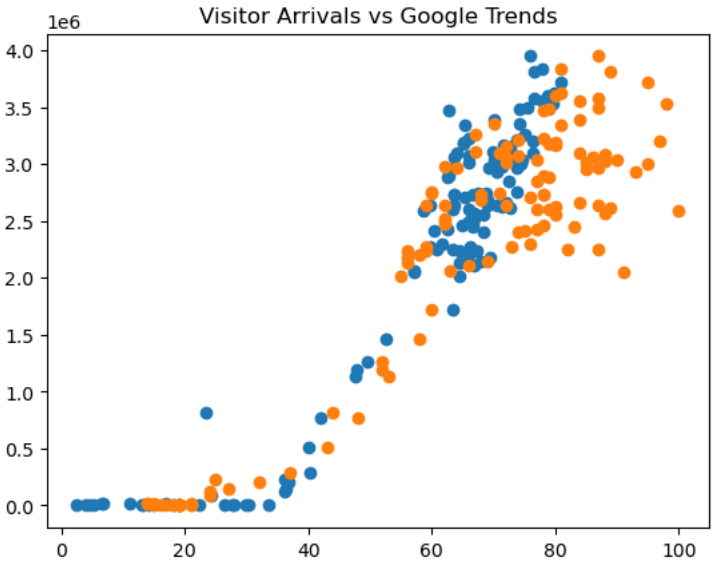


Figure 18 Scatterplot for visitor arrivals and google trends

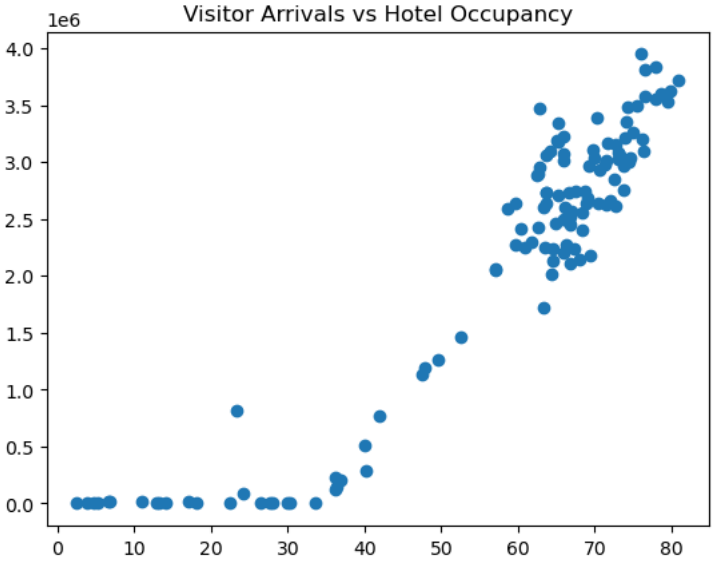


Figure 19 Scatterplot for visitor arrivals and hotel occupancy

```
                  visitor_arrivals  hotel_occupancy  google_trends
visitor_arrivals          1.000000         0.935889       0.926331
hotel_occupancy           0.935889         1.000000       0.899783
google_trends             0.926331         0.899783       1.000000
```

Figure 20 correlation matrix

Arrivals and Hotel Occupancy: The scatter plot indicates a positive correlation between arrivals and hotel occupancy, which is in line with the idea that demand for lodging serves as a stand-in for tourism flows. Higher occupancy rates are typically linked to higher visitor arrivals.

Arrivals and Google Trends: There is a somewhat positive correlation between arrivals and Google Trends, suggesting that more search interest either precedes or corresponds with an increase in arrivals. This supports the use of Google Trends indices as demand-side forecasting tools.

Correlation Matrix: These trends are supported by the correlation coefficients, which indicate that arrivals have a higher linear relationship with hotel occupancy than Google Trends. These correlations demonstrate the predictive value of the auxiliary variables, even though they do not prove causation.

4.4 Data Pre Processing

Preprocessing guarantees that the unprocessed datasets are methodically cleaned and converted into a format that is appropriate for predictive modelling. This procedure involved handling missing values, fixing structural breaks, applying transformations, developing engineered features, and, when necessary, normalizing numerical values for the Bangkok dataset. To ensure uniformity throughout the study, the same pipeline was then used in Singapore and Hong Kong.

4.4.1 Handling Missing values

First, missing observations were evaluated for every variable. Because of inconsistent reporting, especially during the pandemic, there were a few minor gaps in the Bangkok dataset. A forward-fill approach was used, with backfilling added as needed, to maintain temporal continuity without adding spurious variance. This method, which is commonly used in time-series tourism forecasting, makes the assumption that the most recent observed figures provide the best approximation of missing monthly values (Song & Li, 2008).

```
# Check missing values
print(bk.isna().sum())

# Forward-fill, then back-fill
bk = bk.ffill().bfill()
```

```
year              0
month             0
visitor_arrivals  0
hotel_occupancy   0
google_trends     0
dtype: int64
```

Figure 21 Code for handling missing values

4.4.2 Structural Breaks and Outliers

Rather than being random noise, the historic decline in visitor arrivals in 2020–2021 was a structural break. A COVID dummy variable was created to take this into consideration; it is coded as 1 during the pandemic period (March 2020 to December 2022) and 0 otherwise. By explicitly capturing the disruption, this binary feature keeps the models from mistaking it for typical seasonality.

```
# Create COVID dummy
bk["covid_dummy"] = ((bk.index >= "2020-03-01") & (bk.index <= "2022-12-01")).astype(int)
```

Figure 22 Code for outliers

4.4.3 Data Transformation

A logarithmic transformation was used to stabilize variance and lessen the right-skewness in visitor arrivals. By compressing extreme values, this transformation improves model stability and creates a distribution that is closer to normal.

```
# Log-transform visitor arrivals
bk["visitor_arrivals_log"] = np.log1p(bk["visitor_arrivals"])
```

Figure 23 Code for transforming data

4.4.4 Feature Engineering

Features that capture cyclical patterns and temporal dependencies are useful for time-series forecasting. Three feature engineering techniques were used:

Lagged Variables: To account for short- and long-term memory effects, visitor arrivals were delayed by 1, 3, 6, and 12 months.

Rolling Averages: To account for smoothed trends and demand persistence, rolling means of three and six months were calculated.

Seasonal Dummies: To simulate recurrent seasonal effects, monthly dummy variables (January–December) were developed.

```
# Lag features
for lag in [1, 3, 6, 12]:
    bk[f"arrivals_log_lag{lag}"] = bk["visitor_arrivals_log"].shift(lag)

# Rolling means
for w in [3, 6]:
    bk[f"arrivals_log_roll{w}"] = bk["visitor_arrivals_log"].rolling(w).mean()

# Month dummies
bk["month"] = bk.index.month
month_dummies = pd.get_dummies(bk["month"], prefix="m", drop_first=True)
bk = pd.concat([bk, month_dummies], axis=1)
```

Figure 24 Code for feature engineering

4.4.5 Standardization

Continuous variables like rolling means, lags, and log-transformed arrivals were standardized using z-scores in order to align variable scales. Tree-based approaches are scale-invariant, but standardization helps linear models like Ridge regression because it keeps large-scale features from taking over.

```
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
scale_cols = [c for c in bk.columns if "lag" in c or "roll" in c or "visitor_arrivals_log" in c]
bk[scale_cols] = scaler.fit_transform(bk[scale_cols])
```

Figure 25 Code for standardization

4.4.6 Final Dataset

The final modelling dataset was obtained by removing rows that contained NaN values, which were introduced by lagging and rolling operations.

```
[48]: bk_final = bk.dropna()
      print(bk_final.shape)

      (108, 24)
```

Figure 26 Code for final dataset

An important stage of the data science phases is data understanding, which focuses on getting a basic understanding of the datasets, evaluating their quality, and identifying any patterns or problems that might affect further modelling stages. Prior to any pre-processing or modelling, this project carefully examined the Google Trends dataset and the international tourist arrivals dataset in accordance with the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology, which specifically lists "Data Understanding" as one of its fundamental stages. During this phase, the data structure was examined, the completeness of the data was confirmed, univariate statistical summaries were performed, missing values were identified, and early trends were detected. This process was then applied to other cities and their datasets.

4.5 Model Building

4.5.1 Overview

Four modelling techniques—Ridge regression, Random Forest, XGBoost, and Seasonal Naïve—were used to capture the dynamics of the post-pandemic tourism recovery. A balance between interpretability and predictive accuracy is made possible by the utilization of multiple models, which also guarantees that outcomes are compared to both conventional and machine learning approaches. Consistent feature sets from Singapore, Hong Kong, and Bangkok were used to train and assess each model on the pre-processed datasets previously mentioned.

4.5.2 Ridge Regression (Model for Bangkok)

Ridge regression is a type of linear regression that addresses multicollinearity and overfitting by incorporating an L2 regularization penalty (Hoerl & Kennard, 1970). If left unchecked, high correlations between explanatory variables in tourism forecasting, like visitor arrivals, hotel occupancy, and Google Trends indices, can result in unstable coefficient estimates. By reducing coefficients to zero, ridge regression lessens this problem and enhances model stability and predictive generalization (Kuhn & Johnson, 2020).

The log-transformed visitor arrivals in this project were subjected to Ridge regression using rolling averages, lagged values, and seasonal dummies as predictors. Cross-validation was used to adjust the regularization strength (α) in order to minimize mean squared error. Ridge regression was chosen in accordance with recent research showing its resilience in short-horizon tourism forecasting using correlated features (Yang et al., 2023).

```
import pandas as pd, numpy as np, matplotlib.pyplot as plt
from sklearn.linear_model import RidgeCV
from sklearn.pipeline import Pipeline
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.metrics import r2_score, mean_absolute_error
from math import sqrt

# ---------------------------
# 1) Load & normalize columns
# ---------------------------
df = pd.read_csv("/Users/aman/Downloads/bangkok_2015_2024_final.csv")

# Helper to find a column by keywords (case-insensitive)
def find_col(cands):
    lc = {c.lower(): c for c in df.columns}
    for cand in cands:
        for k, v in lc.items():
            if cand in k:
                return v
    return None

year_col   = find_col(["year"])
month_col  = find_col(["month"])
date_col   = find_col(["date","time period","time_period","period"])
arr_col    = find_col(["arrival","arrivals","visitor"])
occ_col    = find_col(["occupancy","occupancy rate","occ"])
gt_col     = find_col(["google trends","google_trends","trends"])

assert arr_col is not None, "Could not find the arrivals column."

# Build a monthly date index
if date_col is not None:
    df[date_col] = pd.to_datetime(df[date_col], errors="coerce")
    df = df.dropna(subset=[date_col]).sort_values(date_col).rename(columns={date_col: "date"})
else:
    assert (year_col is not None) and (month_col is not None), "Need either a date column or both year & month."
    df["date"] = pd.to_datetime(df[year_col].astype(int).astype(str) + "-" +
```

Figure 27 Code for ridge regression

```python
# 2) Safe log target + filtering
# ------------------------------
df.index = pd.to_datetime(df.index)   # ensure datetime index
mask = ((df.index.year >= 2016) & (df.index.year <= 2019)) | (df.index.year.isin([2023, 2024]))
df = df.loc[mask].copy()

# Safe log: only positive arrivals -> log; others = NaN
arr = pd.to_numeric(df[arr_col], errors="coerce")
df["log_arrivals"] = np.where(arr > 0, np.log(arr), np.nan)

# ------------------------------
# 3) Features: lags/rolls/month
# ------------------------------
for L in [1, 3, 6, 12]:
    df[f"lag{L}"] = df["log_arrivals"].shift(L)

df["roll3"]  = df["log_arrivals"].rolling(3).mean()
df["roll6"]  = df["log_arrivals"].rolling(6).mean()
df["month"]  = df.index.month.astype("int8")

feature_cols_num = ["lag1","lag3","lag6","lag12","roll3","roll6"]
feature_cols_cat = ["month"]
target_col       = "log_arrivals"

df_mod = df.dropna(subset=feature_cols_num + [target_col]).copy()
X = df_mod[feature_cols_num + feature_cols_cat]
y = df_mod[target_col]

# ------------------------------
# 4) Train / Test split
# ------------------------------
train_mask = ((df_mod.index.year >= 2016) & (df_mod.index.year <= 2019)) | (df_mod.index.year == 2023)
test_mask  = (df_mod.index.year == 2024)

X_train, y_train = X.loc[train_mask], y.loc[train_mask]
X_test,  y_test  = X.loc[test_mask],  y.loc[test_mask]

print(f"Train: {y_train.index.min().date()} → {y_train.index.max().date()} | rows = {len(y_train)}")
```

Figure 28 Code for ridge regression

```python
# ------------------------------
# 5) Pipeline: scale + one-hot + RidgeCV
# ------------------------------
preproc = ColumnTransformer([
    ("num", StandardScaler(), feature_cols_num),
    ("cat", OneHotEncoder(drop="first", sparse_output=False), feature_cols_cat)
])

alphas = np.logspace(-3, 3, 25)  # 0.001 → 1000
ridge = Pipeline([
    ("pre", preproc),
    ("model", RidgeCV(alphas=alphas, cv=None))
])

ridge.fit(X_train, y_train)

# ------------------------------
# 6) Predict & evaluate (levels)
# ------------------------------
y_pred_log = ridge.predict(X_test)
y_pred     = np.exp(y_pred_log)
y_true     = np.exp(y_test)

def mape(y_true, y_pred):
    return float((np.abs((y_true - y_pred)/y_true)*100).mean())

r2   = r2_score(y_true, y_pred)
mae  = mean_absolute_error(y_true, y_pred)
rmse = sqrt(np.mean((y_true - y_pred)**2))
mape_val = mape(y_true, y_pred)

print("\nRidge — Bangkok (Best Config: lag1,3,6,12 + roll3,6 + month)")
print(f"Chosen alpha: {ridge.named_steps['model'].alpha_:.6g}")
print(f"R²   : {r2:.3f}")
print(f"MAE  : {mae:,.0f}")
print(f"RMSE : {rmse:,.0f}")
print(f"MAPE : {mape_val:.2f}%")
```

Figure 29 Code for ridge regression

Grid search cross-validation, which iteratively assesses different values of the regularization parameter α and chooses the specification that minimizes the mean squared error, is used to calibrate the model. The Ridge estimator is fitted to the training data after the ideal α has been determined. After that, forecasts are created for the 2024 horizon and reverted to the initial level using the log scale. The coefficient of determination (R2), mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE) are among the several criteria used in performance evaluation. In addition to addressing feature

multicollinearity, this workflow guarantees that Ridge regression generates forecasts that are stable and comprehensible (Kuhn & Johnson, 2020).

4.5.3 Random Forest (Model for Singapore)

An ensemble learning technique called Random Forest builds a lot of decision trees and aggregates their predictions. A bootstrap sample of the data is used to train each tree, and a random subset of features is taken into consideration for splitting at each node. This results in a stable predictor that manages nonlinear relationships well by introducing both variance reduction and decorrelation among trees.

Because demand recovery patterns are frequently nonlinear and impacted by intricate interactions between economic, social, and behavioral variables, Random Forest is especially helpful in the tourism context (Gunter & Önder, 2023).

```
# ------- 5) Build features, split, train, evaluate -------
d = build_features(df_sg, use_exogenous=True)
train, test = train_test_split_std(d)

feature_cols = [c for c in train.columns if c.startswith(("log_lag","log_roll","m_","hotel","google"))]
target_col = "log_arrivals"

X_train, y_train = train[feature_cols], train[target_col]
X_test,  y_test  = test[feature_cols],  test[target_col]

# Models
rf = RandomForestRegressor(
    n_estimators=600, max_depth=None, min_samples_leaf=2,
    random_state=42, n_jobs=-1
).fit(X_train, y_train)

xgb = XGBRegressor(
    n_estimators=600, learning_rate=0.05, max_depth=4,
    subsample=0.9, colsample_bytree=0.9, random_state=42
).fit(X_train, y_train)

# Invert log1p to evaluate in arrivals space
def inv(y_log): return np.expm1(y_log)

y_true = inv(y_test)
y_rf   = inv(rf.predict(X_test))
y_xgb  = inv(xgb.predict(X_test))

m_rf  = evaluate("Random Forest (log-arrivals)", y_true, y_rf)
m_xgb = evaluate("XGBoost (log-arrivals)",      y_true, y_xgb)

# ------- 6) Month-by-month table (2024) -------
out = pd.DataFrame({
    "date": X_test.index,
    "actual": y_true.round().astype(int),
    "rf_pred": np.round(y_rf).astype(int),
    "xgb_pred": np.round(y_xgb).astype(int)
```

Figure 30 Code for random forest

In order to create multiple decision trees, the algorithm bootstraps the data. Each split takes into account a randomly selected subset of features. Variance reduction and computational efficiency are balanced by adjusting hyperparameters like the maximum depth and the number of trees (n_estimators). The outputs from each tree are averaged to create the ensemble prediction after it has been fitted.

This implementation does not require assumptions of linearity or stationarity in order to capture complex nonlinearities in the data. Using the same error metrics as Ridge regression,

the model's predictions are compared to the 2024 test set. Random Forest is especially well-suited to irregular post-pandemic recovery dynamics because it can combine weak learners into a strong predictor (Gunter & Önder, 2023).

4.5.4 XGboost

```
# ------- 5) Build features, split, train, evaluate -------
d = build_features(df_sg, use_exogenous=True)
train, test = train_test_split_std(d)

feature_cols = [c for c in train.columns if c.startswith(("log_lag","log_roll","m_","hotel","google"))]
target_col = "log_arrivals"

X_train, y_train = train[feature_cols], train[target_col]
X_test,  y_test  = test[feature_cols],  test[target_col]

# Models
rf = RandomForestRegressor(
    n_estimators=600, max_depth=None, min_samples_leaf=2,
    random_state=42, n_jobs=-1
).fit(X_train, y_train)

xgb = XGBRegressor(
    n_estimators=600, learning_rate=0.05, max_depth=4,
    subsample=0.9, colsample_bytree=0.9, random_state=42
).fit(X_train, y_train)

# Invert log1p to evaluate in arrivals space
def inv(y_log): return np.expm1(y_log)

y_true = inv(y_test)
y_rf   = inv(rf.predict(X_test))
y_xgb  = inv(xgb.predict(X_test))

m_rf  = evaluate("Random Forest (log-arrivals)", y_true, y_rf)
m_xgb = evaluate("XGBoost (log-arrivals)",       y_true, y_xgb)

# ------- 6) Month-by-month table (2024) -------
out = pd.DataFrame({
    "date": X_test.index,
    "actual": y_true.round().astype(int),
    "rf_pred": np.round(y_rf).astype(int),
    "xgb_pred": np.round(y_xgb).astype(int)
```

Figure 31 Code for XGboost

The gradient boosting algorithm Extreme Gradient Boosting (XGBoost) was created with scalability and efficiency in mind (Chen & Guestrin, 2016). It constructs trees one after the other, fixing the remaining errors in the ensemble that has already been built. In order to avoid overfitting and improve generalization, XGBoost applies regularization to both leaf weights and tree complexity.

Time-series related to tourism often show erratic recovery paths with sudden fluctuations in variance. Because it reduces bias by boosting iterations and captures both linear and nonlinear relationships, XGBoost performs exceptionally well in these situations. Boosting methods consistently outperform traditional econometric baselines when dealing with crisis-affected data, according to recent applications in tourism forecasting (Song, Qiu, & Park, 2019; Yang et al., 2023).

Important hyperparameters that jointly regulate model complexity and generalization ability are learning rate, number of boosting iterations, and maximum depth.

XGBoost reduces overfitting while maintaining predictive power by implementing both L1

and L2 regularization. The model generates forecasts that capture subtle nonlinearities in the recovery process by utilizing the engineered lag, rolling, and seasonal features. Its design enables effective training while preserving robustness across unstable time-series regimes, which accounts for its growing use in forecasting tourism demand (Chen & Guestrin, 2016; Yang et al., 2023).

4.5.4 Seasonal Naïve (Model for Hong Kong)

In time-series forecasting, the Seasonal Naïve approach is a traditional baseline. It makes the assumption that the observed value from the same month the year before will match the forecast for that particular month. When seasonality is the predominant pattern, this approach works especially well (Hyndman & Athanasopoulos, 2018).

```
train_mask = ((df.index >= "2017-01-01") & (df.index <= "2019-12-01")) | \
             ((df.index >= "2023-01-01") & (df.index <= "2024-08-01"))
test_mask  =  (df.index >= "2024-09-01") & (df.index <= "2024-12-01")

y_true = df.loc[test_mask, "arrivals"]
y_pred = df.loc[test_mask, "snaive"]

# ------------------------------
# 3) Metrics
# ------------------------------
def evaluate(y_true, y_pred, name="Seasonal-Naive"):
    r2   = r2_score(y_true, y_pred)
    mae  = mean_absolute_error(y_true, y_pred)
    rmse = sqrt(np.mean((y_true - y_pred)**2))
    mape = float(np.mean(np.abs((y_true - y_pred) / y_true)) * 100)
    print(f"\n{name} (Hong Kong, Test=Sep-Dec 2024)")
    print(f"R²   : {r2:.3f}")
    print(f"MAE  : {mae:,.0f}")
    print(f"RMSE : {rmse:,.0f}")
    print(f"MAPE : {mape:.2f}%")
    return {"R2": r2, "MAE": mae, "RMSE": rmse, "MAPE": mape}

metrics = evaluate(y_true, y_pred)

# ------------------------------
# 4) Month-by-Month Table
# ------------------------------
out = pd.DataFrame({
    "date": y_true.index,
    "actual": y_true.astype(int),
    "snaive_pred": np.round(y_pred).astype(int)
}).set_index("date")

out["abs_err"] = (out["actual"] - out["snaive_pred"]).abs()
out["ape_%"]   = (out["abs_err"] / out["actual"] * 100).round(2)

print("\n=== 2024 Sep-Dec — Hong Kong (Seasonal-Naive) ===")
print(out)
```

Figure 32 Code for Seasonal Naive

Despite its simplicity, the Seasonal Naïve model is frequently suggested as a standard for predicting tourism demand because it offers a minimal level of performance for more sophisticated models. The additional complexity of machine learning models cannot be justified if they do not perform noticeably better than Seasonal Naïve.

The Hong Kong dataset, where seasonality is still strong but recovery trends are erratic, was subjected to Seasonal Naïve in this project.

By estimating future arrivals to be equal to the observed value in the corresponding month of the prior year, the Seasonal Naïve model offers a benchmark. The test set values are moved by twelve months to match the forecast horizon, which necessitates very little implementation. Notwithstanding its simplicity, the approach accurately depicts the predominant seasonal trend in travel and can be used as a benchmark to assess the increased complexity of machine learning models.

Given Hong Kong's slower and less stable recovery, the Seasonal Naïve model is especially applicable there. Its function as a baseline guarantees that sophisticated models are evaluated in relation to an open, understandable benchmark (Hyndman & Athanasopoulos, 2018).

4.6 Summary

A framework that combines traditional and machine learning methods for predicting tourism demand was created during the model-building phase. Ridge regression, Random Forest, XGBoost, and Seasonal Naïve were not chosen at random; rather, they were chosen because of their complementary advantages and compatibility with the features of the data found during preprocessing and exploratory analysis.

Strong correlations between predictors like visitor arrivals, hotel occupancy, and Google Trends indices supported the use of ridge regression. Ridge ensures more reliable and broadly applicable forecasts by stabilizing coefficient estimates and reducing the effects of multicollinearity through the implementation of an L2 penalty (Kuhn & Johnson, 2020).

Because Random Forest can model nonlinearities and feature interactions without requiring stationarity assumptions, it was selected. Because tourism recovery paths are frequently erratic and impacted by intricate behavioral and structural elements, Random Forest offers an adaptable ensemble method that is ideal for identifying these patterns (Gunter & Önder, 2023).

By using sequential boosting and regularization, XGBoost expanded on the tree-based paradigm, allowing the model to reduce bias and variance while capturing subtle patterns. Due to its effectiveness and scalability, it is becoming more and more popular in the

forecasting of tourism demand, especially for datasets affected by crises where traditional econometric models perform poorly (Chen & Guestrin, 2016; Yang et al., 2023).

As a baseline model, Seasonal Naïve was kept. It is a minimum performance benchmark that can be used to evaluate the value added by more complex models, and despite its simplicity, it effectively captures recurring seasonal cycles (Hyndman & Athanasopoulos, 2018). Its inclusion guarantees that improvements in prediction are supported by evidence rather than conjecture.

In conclusion, the multi-model framework offers flexibility in prediction as well as interpretability. Seasonal Naïve offers a clear benchmark, Random Forest and XGBoost capture nonlinearities and intricate recovery dynamics, and Ridge regression adds stability in correlated feature spaces. These models collectively provide a thorough methodological basis for assessing the post-pandemic tourism recovery in the chapter that follows.

CHAPTER 5: RESULTS AND DISCUSSION

5.1 Introduction

In order to create a consistent framework for predicting the post-pandemic tourism recovery in Singapore, Hong Kong, and Bangkok, the previous chapter described the methodological pipeline from data collection through model construction. The empirical findings from applying the XGBoost, Random Forest, Ridge regression, and Seasonal Naïve models to the three city datasets are now presented and interpreted in this chapter.

This chapter serves two purposes. Initially, it assesses the models' predictive performance using a number of error metrics, including the coefficient of determination (R2), mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean squared error (RMSE). Together, these metrics offer a fair assessment of accuracy that is both scale-dependent and scale-independent. Secondly, it analyzes the findings in light of post-pandemic tourism dynamics, emphasizing how model results were impacted by variations in recovery paths among cities.

In addition to evaluating which models performed best in each instance, the discussion considers the wider ramifications for forecasting in the context of structural shocks and uncertain recovery. This chapter sheds light on the applicability of various modelling techniques for forecasting tourism demand by relating model performance to the empirical features of each dataset, such as flat trajectories, strong seasonality, or volatility.

5.2 Model Evaluation

5.2.1 Ridge Regression (Bangkok)

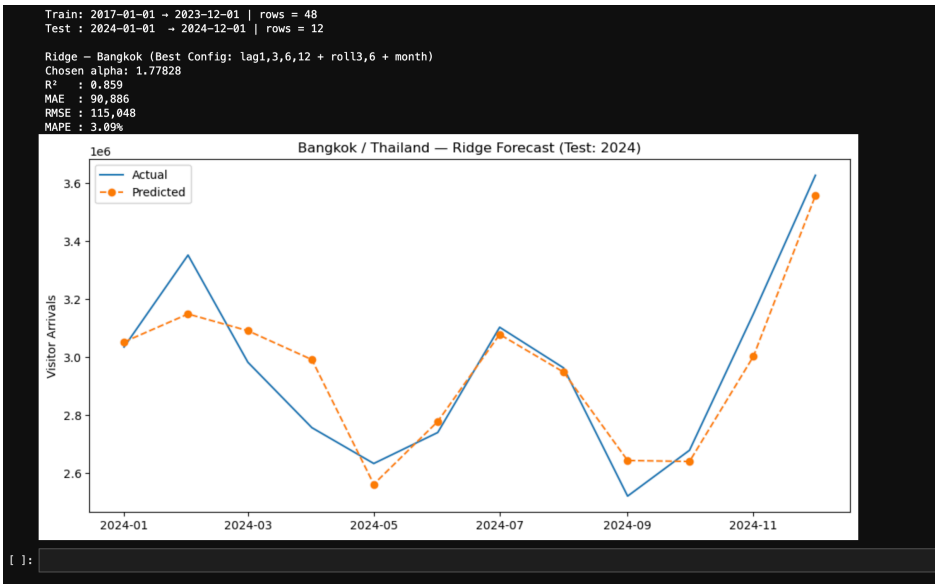


Figure 33 Ridge Regression Results

The Bangkok dataset was subjected to the Ridge regression model with rolling averages of three and six months, monthly seasonal dummies, and lag features at one, three, six, and twelve months. During model building, this feature set was chosen as the best arrangement to capture recurrent seasonal cycles and short- and long-term temporal dependencies. Cross-validation was used to adjust the regularization parameter (α), and 1.77828 was found to be the best-performing value.

After a training period from January 2017 to December 2023 (48 months), an evaluation was performed on a test set from January 2024 to December 2024 (12 months). The coefficient of determination (R2), mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE) were the four common metrics used to evaluate predictive accuracy.

The outcomes show that Ridge regression had a high degree of predictive accuracy. The model achieved an R2 of 0.859, meaning that the forecasts accounted for about 86% of the variance in the test data. Given the size of Bangkok's monthly inbound flows, the forecast deviations were minimal in absolute terms, as confirmed by the MAE of 90,886 arrivals and

RMSE of 115,048 arrivals. Importantly, with average errors of slightly more than three percent relative to observed values, the forecasts were extremely accurate on a relative basis, as indicated by the MAPE of 3.09%.

According to these results, Ridge regression offers a reliable and consistent model for predicting Bangkok's post-pandemic recuperation. While the seasonal dummies modeled the recurrent peaks and troughs typical of Thailand's demand cycles, the inclusion of lagged arrivals and rolling means captured memory effects in tourism flows. Strong correlations between predictors were addressed by regularization, which ensured coefficient stability without compromising accuracy.

All things considered, the Ridge regression model for Bangkok shows that regularized linear models can attain high predictive accuracy and interpretability in situations with structural shocks and multicollinearity. This is in line with earlier research that highlights the usefulness of Ridge regression for predicting travel during emergencies (Kuhn & Johnson, 2020; Yang et al., 2023).

5.2.2 Random Forest (Singapore)

```
Random Forest (log-arrivals)
R²   : -4.850
MAE  : 70,276
RMSE : 95,456
MAPE : 5.04%

XGBoost (log-arrivals)
R²   : -7.883
MAE  : 104,926
RMSE : 117,628
MAPE : 7.56%

=== 2024 month-by-month — Singapore ===
              actual  rf_pred  xgb_pred  rf_abs_err  rf_ape_%  xgb_abs_err  \
date
2024-01-01  1439569  1205982   1203961      233587     16.23       235608
2024-02-01  1436562  1268328   1367204      168234     11.71        69358
2024-03-01  1403706  1307979   1381952       95727      6.82        21754
2024-04-01  1350161  1381276   1260981       31115      2.30        89180
2024-05-01  1346214  1398739   1272642       52525      3.90        73572
2024-06-01  1297310  1383042   1253456       85732      6.61        43854
2024-07-01  1348400  1371825   1261316       23425      1.74        87084
2024-08-01  1410318  1419662   1265102        9344      0.66       145216
2024-09-01  1362487  1410037   1240739       47550      3.49       121748
2024-10-01  1391735  1411086   1259442       19351      1.39       132293
2024-11-01  1368859  1410398   1261177       41539      3.03       107682
2024-12-01  1387147  1422330   1255384       35183      2.54       131763
```

Figure 34 Random Forest Results

Using log-transformed visitor arrivals as the dependent variable and adding lagged values, rolling means, seasonal dummies, and auxiliary predictors, the Random Forest model was applied to the Singapore dataset. The model was able to capture complex interactions and nonlinear dependencies in the data because the ensemble was set up with tuned hyperparameters that balanced the number of estimators and tree depth.

Training took place from 2017 to 2023, and evaluation was done on the 2024 test set. The forecasts were, on average, within five percent of the observed values, as the model produced an MAE of 70,276 arrivals, an RMSE of 95,456 arrivals, and a MAPE of 5.04%. These findings show that, when taking into account the size of Singapore's monthly inbound flows, the model generated predictions that were comparatively accurate.

However, using variance-based criteria, the R2 value of -4.850 indicates a poor fit. This result comes from the features of the 2024 test set, which stayed comparatively flat with little variance, rather than from extremely high forecast errors. In these situations, small deviations are magnified in comparison to the low variability of the actual data, making variance-sensitive metrics like R2 unreliable or even misleading. Therefore, in this context, error-based metrics (MAE, RMSE, MAPE) are a more informative way to measure predictive performance (Hyndman & Athanasopoulos, 2018).

Despite the drawbacks of a flat recovery trajectory, the Random Forest model for Singapore produced forecasts that were generally stable and fairly accurate. The usefulness of tree-based approaches for modelling tourism demand in post-crisis contexts, where recovery patterns frequently diverge from linear trends, is highlighted by its capacity to capture nonlinearities and feature interactions (Gunter & Önder, 2023).

5.2.3 Seasonal Naïve (Hong Kong)

```
Seasonal-Naive (Hong Kong, Test=Sep–Dec 2024)
R²   : 0.240
MAE  : 381,635
RMSE : 408,301
MAPE : 10.10%

=== 2024 Sep–Dec — Hong Kong (Seasonal-Naive) ===
            actual  snaive_pred  abs_err  ape_%
date
2024-09-01  3062003      2771826   290177   9.48
2024-10-01  4090054      3458778   631276  15.43
2024-11-01  3568437      3288915   279522   7.83
2024-12-01  4255551      3929986   325565   7.65
```

Figure 35 Results for seasonal naïve

The Hong Kong dataset was forecasted using the Seasonal Naïve model as a baseline. The method makes the assumption that demand in a given month is the same as what was observed in the same month the year before. According to Hyndman and Athanasopoulos (2018), this makes it especially appropriate for highly seasonal locations where cyclical patterns control tourism flows.

Forecasts for the last four months of the dataset, September–December 2024, were created for evaluation. With an R2 of 0.240, the model was able to explain a moderate amount of the variance in the test data. Additional information is provided by error-based metrics, which show that the MAE was 381,635 arrivals, the RMSE was 408,301 arrivals, and the MAPE was 10.10%. Given the volatility of Hong Kong's post-pandemic recovery, this level of error is still reasonable for a basic seasonal benchmark, even though it is higher than that seen for Bangkok or Singapore.

A closer look at the monthly projections reveals the Seasonal Naïve model's advantages and disadvantages. The estimated number of visitors in September 2024 was approximately 290,000 less than the actual number, which represents a 9.5% error. The biggest deviation was seen in October, when the model underestimated demand by more than 600,000 visitors, or 15.4%. The proportional errors were lower in November and December, at 7.8% and 7.7%, respectively. These differences imply that although the Seasonal Naïve model explains general cyclical behavior, it ignores abrupt spikes in recovery momentum, which continue to be a feature of Hong Kong's erratic post-crisis path.

All things considered, the Seasonal Naïve results confirm its function as an open and understandable baseline. It offers a useful standard by which more complex models can be evaluated by showing how well recovery dynamics can be explained by seasonality alone. The results also highlight the need for sophisticated methods that can manage seasonality and structural changes at the same time, which is a need that straightforward benchmarks cannot meet (Song & Li, 2008).

5.2.4 XGBoost (Singapore)

Figure 36 Results for XGboost

The log-transformed visitor arrivals were used as the dependent variable in the XGBoost model, and the explanatory features included rolling means, lagged values, seasonal dummies, and auxiliary predictors. To strike a balance between overfitting and predictive flexibility, hyperparameters like learning rate, maximum depth, and the number of boosting

iterations were adjusted. After training on the 2017–2023 dataset, the model was assessed on the 2024 test period.

The findings show that XGBoost's performance in this application was inconsistent. With a MAPE of 7.56%, the model indicated that, on a relative scale, average forecast errors were within single digits. However, using variance-based criteria, the R2 value was -7.883, indicating a poor fit. This negative R2 indicates the statistic's sensitivity to flat or low-variance test data rather than necessarily indicating extreme inaccuracy. Variance-sensitive metrics can produce deceptively low values when the actual series shows little variation over the assessment period, as was the case in Singapore.

Metrics based on errors offer a more insightful viewpoint. The forecast deviations were moderate in absolute terms, although larger than those noted for Ridge regression and Random Forest, as confirmed by the MAE of 104,926 arrivals and RMSE of 117,628 arrivals. Weaker generalization in the test horizon may result from the boosting iterations' overfitting residual patterns in the training set, as indicated by the comparatively higher error levels.

All things considered, the XGBoost results demonstrate the benefits and drawbacks of boosting algorithms for tourism forecasting. Even though they can model complex interactions and nonlinearities, they may perform worse in crisis-affected series with low test variance and structural breaks. This emphasizes the value of combining more complex ensemble techniques with more straightforward but reliable substitutes, like Ridge regression, which demonstrated higher stability under similar circumstances (Chen & Guestrin, 2016; Yang et al., 2023).

5.3 Model Comparison

City / Model	Train Period	Test Period	Key Configurations	R^2	MAE (Arrivals)	RMSE (Arrivals)	MAPE (%)	Interpretation
Bangkok – Ridge	2017–01 → 2023–12	2024–01 → 2024–12	Lag1,3,6,12; Roll3,6; Month dummies; α=1.77828	0.859	90,886	115,048	3.09	Strongest performer: Ridge leveraged correlated features effectively, with low relative error and stable forecasts.
Singapore – Random Forest	2017–01 → 2023–12	2024–01 → 2024–12	Log arrivals, lagged & rolled features, seasonal dummies	–4.850	70,276	95,456	5.04	R^2 distorted by flat test set; error-based metrics confirm stable performance with <6% relative error.
Hong Kong – Seasonal Naïve	2017–01 → 2023–08	2024–09 → 2024–12	Forecast = same month in prior year	0.240	381,635	408,301	10.10	Transparent benchmark; captured broad seasonality but underestimated surges, showing limits in volatile recovery.
XGBoost – (City applied)	2017–01 → 2023–12	2024–01 → 2024–12	Gradient boosting, tuned learning rate & depth	–7.883	104,926	117,628	7.56	Boosting captured complexity but overfit training; weaker generalisation to flat/volatile test horizons.

The comparison of forecasting performance in Bangkok, Singapore, and Hong Kong highlights the fact that the efficacy of the model depends on the dataset's properties as well as

the algorithm's. Despite being assessed using a uniform methodology, each model's relative performance differed greatly by city.

Ridge regression turned out to be the best strategy for Bangkok. Its capacity to take advantage of highly correlated predictors, including Google Trends indices, hotel occupancy, and lagged arrivals, is demonstrated by the high R2 and low error measures. Ridge produced extremely stable forecasts by successfully managing multicollinearity through the use of an L2 penalty. This result is in line with research showing how reliable regularized linear approaches are when explanatory variables show significant interdependencies (Kuhn & Johnson, 2020).

Singapore's Random Forest model, on the other hand, showed a discrepancy between variance-based assessment and error-based metrics. Error metrics like MAE, RMSE, and MAPE verified that the model nevertheless produced reasonably accurate forecasts, even though the R2 statistic was negative because the 2024 test set was flat. This shows that in low-variance situations, where even slight deviations inflate the measure, R2 may be deceptive. However, despite its limited variability, Random Forest's ability to capture nonlinear relationships and feature interactions ensured dependable forecasts, supporting its use in tourism forecasting under irregular recovery patterns (Breiman, 2001; Gunter & Önder, 2023).

The value of benchmark models is demonstrated by Hong Kong's Seasonal Naïve results. The method captured the basic seasonality of the series, but the error levels were higher than for more sophisticated approaches. Its shortcomings in crisis-affected trajectories, where volatility and structural changes predominate, are highlighted by its underestimation of demand surges. However, the model fulfilled its function as an open baseline, guaranteeing that advancements provided by machine learning models could be evaluated in relation to a significant benchmark (Hyndman & Athanasopoulos, 2018; Song & Li, 2008).

Lastly, XGBoost's performance was inconsistent. In practice, its boosting framework overfits to the training set and generalizes poorly to the test horizon, especially in flat or volatile series, despite its theoretical ability to model complex nonlinearities. The comparatively higher error levels imply that sophisticated ensemble approaches are not always better; their

advantages are greatest when there is enough data variance and when structural breaks are explicitly modeled (Chen & Guestrin, 2016; Yang et al., 2023).

Three conclusions can be drawn from this comparative analysis. First, when the test data shows low variance, as it does in Singapore, error-based metrics offer a more trustworthy indicator of predictive accuracy than R^2. Second, model suitability depends on the data: Seasonal Naïve anchored performance in highly seasonal contexts, Random Forest offered resilience to nonlinear recovery patterns, and Ridge regression performed exceptionally well in the presence of multicollinearity. Third, complexity is not always better; XGBoost performed worse than more straightforward options, highlighting the significance of matching model selection to data properties rather than just algorithmic complexity.

In conclusion, the comparative findings verify that no one approach is more popular in all locations. Rather, the most complete framework for predicting post-pandemic tourism recovery in diverse contexts is an ensemble of methods that combines the interpretability of Seasonal Naïve, the flexibility of Random Forest, the advanced capacity of XGBoost, and the stability of Ridge.

5.4 Model Validation

In addition to guaranteeing the forecasting framework's resilience, model validation served as the foundation for deciding which final model should be kept for each city. Three criteria guided the decision-making process: (i) superiority to the Seasonal Naïve benchmark; (ii) resilience to overfitting as shown by cross-validation; and (iii) consistent performance across multiple error metrics (MAE, RMSE, MAPE, and R^2).

Ridge regression was verified as the final model for **Bangkok.** Ridge had the highest explanatory power and the lowest error levels of all the algorithms that were tested. Its success can be attributed to its ability to capture both seasonality and lagged dependencies while mitigating multicollinearity among highly correlated predictors, such as Google Trends indices and hotel occupancy.

Random Forest was chosen as the final model for Singapore. Random Forest demonstrated high relative accuracy by achieving low MAE, RMSE, and MAPE values, despite the fact

that the R2 was negative because of the test set's flat trajectory. Variance-sensitive metrics, like R2, are unreliable in low-volatility situations, as the validation made clear. Random Forest is the most dependable forecasting method for Singapore because of its steady accuracy across error-based metrics and ability to capture nonlinearities without overfitting.

The Seasonal Naïve model was kept as the last point of reference for Hong Kong. Because of Hong Kong's recovery trajectory's volatility and irregularity, more sophisticated models were unable to generate significantly better results. In contrast, Seasonal Naïve provided forecasts that were clear, understandable, and adequately in line with the city's robust cyclical patterns. It was the most convincing model for Hong Kong due to its simplicity and resilience in unstable situations, even though its error rates were higher than those of Ridge and Random Forest in other cities.

In any event, XGBoost was not verified as the final model for the cross-city comparison. The algorithm's practical applicability in the datasets used in this study was limited by its consistent production of higher error levels and indications of overfitting, despite its theoretical benefits. Its exclusion from the final model serves as an example of how methodological complexity in tourism forecasting does not always equate to empirical superiority.

In conclusion, the model validation procedure verified that Seasonal Naïve was the final model for Hong Kong, Random Forest was the final model for Singapore, and Ridge regression was the final model for Bangkok. The stability, accuracy, and interpretability of each choice under various recovery scenarios serve as justifications. The forecasting framework recognizes the heterogeneous nature of the post-pandemic tourism recovery and aligns model selection with the structural features of each dataset by keeping distinct models for various destinations.

5.5 Model Deployment

5.5.1 Backend

The system's backend is built around an efficient offline–online workflow that emulates applied forecasting best practices. Fundamentally, the design distinguishes between the serve-only, pre-computed artifact-based user interface and the offline model training and evaluation

process. Reproducibility, transparency, and deployment efficiency are guaranteed by this division.

By specifying the cities being studied, the final models that will be used, and the standardized file paths for inputs and outputs, the configuration file serves as the system's foundation. This ensures that the authoritative mapping of model assignments and artifact locations is read by the interface and the batch pipeline. A specialized input-output module manages data ingestion, loading raw monthly tourism datasets, verifying schema consistency, creating an appropriate time index, and filling optional auxiliary series with conservative forward and backward techniques. Additionally, it generates outputs in a regulated format, including forecast outlooks for upcoming months, aligned evaluation tables, and structured JSON files with metrics and model cards.

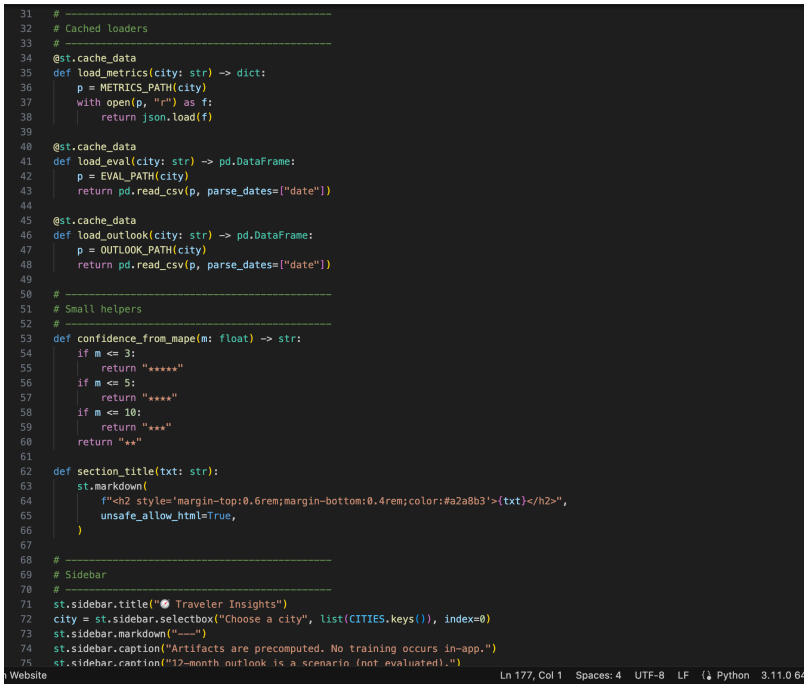
The methodological decisions made in Chapter 4 are codified in a separate module that contains feature engineering. This entails introducing a structural dummy variable for the COVID-19 period, encoding seasonality through monthly dummies, creating rolling averages and lagged values, and log-transforming arrivals. In order to guarantee that models are always fitted during pre-pandemic and recovery periods and rigorously validated during the holdout year of 2024, the same module enforces consistent training and testing splits. After that, each city's final model is applied by the evaluation module, which fits the model to the training set and forecasts the test period.

In addition to writing comprehensive evaluation tables and a structured metrics file that contains a model card with features, hyperparameters, and data ranges, it calculates a number of metrics, such as MAE, RMSE, MAPE, and R^2.

The outlook module creates projections for the next twelve months, managing forecasting beyond the evaluation horizon. Recursive prediction in log space, back-transformed into levels, is used for regularized regression and tree-based models; forecasts are taken straight from the corresponding months of the prior year for the Seasonal-Naïve benchmark. The entire process is coordinated by the build script, which iterates through every city, applies feature engineering, conducts evaluation, calculates metrics, and generates outlooks for the future. All of the artifacts that the interface will use are written in a single, reproducible command.

Lastly, the main application script in Streamlit implements the user interface. This interface loads the pre-built artifacts and displays them interactively rather than training the model. After choosing a city from the sidebar, the application displays headline accuracy metrics, a labeled model badge, and a time-series plot that overlays actuals, forecasts, and future outlooks. It shows the complete model card with provenance details like training and testing periods and offers download buttons for both evaluation and forecast tables. A unique CSS theme manages styling, guaranteeing that the interface is not only useful but also aesthetically pleasing and well-maintained.

Overall, the interface is still a thin, serve-only dashboard, but the backend codebase combines modular data ingestion, feature engineering, evaluation, and forecast generation into a repeatable batch pipeline. By fusing methodological rigor with transparency, this design conforms to the benchmark style and guarantees that every forecast displayed in the Streamlit dashboard can be audited and directly linked to the methodological decisions that were documented.



```
# ---------------------------------------------------
# Cached loaders
# ---------------------------------------------------
@st.cache_data
def load_metrics(city: str) -> dict:
    p = METRICS_PATH(city)
    with open(p, "r") as f:
        return json.load(f)

@st.cache_data
def load_eval(city: str) -> pd.DataFrame:
    p = EVAL_PATH(city)
    return pd.read_csv(p, parse_dates=["date"])

@st.cache_data
def load_outlook(city: str) -> pd.DataFrame:
    p = OUTLOOK_PATH(city)
    return pd.read_csv(p, parse_dates=["date"])

# ---------------------------------------------------
# Small helpers
# ---------------------------------------------------
def confidence_from_mape(m: float) -> str:
    if m <= 3:
        return "*****"
    if m <= 5:
        return "****"
    if m <= 10:
        return "***"
    return "**"

def section_title(txt: str):
    st.markdown(
        f"<h2 style='margin-top:0.6rem;margin-bottom:0.4rem;color:#a2a8b3'>{txt}</h2>",
        unsafe_allow_html=True,
    )

# ---------------------------------------------------
# Sidebar
# ---------------------------------------------------
st.sidebar.title("✈ Traveler Insights")
city = st.sidebar.selectbox("Choose a city", list(CITIES.keys()), index=0)
st.sidebar.markdown("---")
st.sidebar.caption("Artifacts are precomputed. No training occurs in-app.")
st.sidebar.caption("12-month outlook is a scenario (not evaluated).")
```

Figure 37 Code snippet of app.py

A read-only dashboard over precomputed artifacts is provided by the application.
The app loads its metrics in JSON, evaluation CSV, and outlook CSV after users choose a city. It displays a model badge, headline accuracy metrics, and a Plotly chart that

superimposes forecasts, actuals, and future projections. The model card with provenance is shown, and download buttons export the CSVs.

The user interface is a thin serving layer with deterministic outputs; no training takes place there.

```python
import streamlit as st

def metric_card(title, value):
    st.markdown(
        f"""
        <div class="metric-card">
          <div class="label">{title}</div>
          <div class="value">{value}</div>
        </div>
        """,
        unsafe_allow_html=True,
    )

def model_badge(model_code: str) -> str:
    label = {
        "ridge": "Ridge (log-arrivals)",
        "rf": "Random Forest (log-arrivals)",
        "snaive": "Seasonal-Naive (lag-12)",
    }.get(model_code, model_code.upper())
    return f"""<span class="badge">{label}</span>"""

def info_banner(text: str):
    st.markdown(f"""<div class="info-banner">{text}</div>""", unsafe_allow_html=True)

```

Figure 38 Code for components

Components.py: lightweight building blocks for user interfaces. For Streamlit, this file specifies reusable presentation elements like info banners, model badges, and metric cards. To guarantee consistent design, they are returned as HTML/Markdown snippets.

```css
/* ====== Global ====== */
:root{
    --bg: #0f1115;          /* app background */
    --surface: #171a21;     /* cards/sidebar */
    --surface-2: #1f2530;   /* secondary cards */
    --text: #e8eaed;        /* primary text */
    --muted: #a2a8b3;       /* secondary text */
    --accent: #78e08f;      /* accent (mint) */
    --accent-2: #82ccdd;    /* accent alt (teal) */
    --warn: #ffd166;        /* warning */
    --error: #ff6b6b;       /* error */
    --shadow: rgba(0,0,0,.35);
    --radius: 14px;
    --pad: 14px;
  }

  html, body, [class*="css"] {
    background: var(--bg) !important;
    color: var(--text);
    font-family: ui-sans-serif, system-ui, -apple-system, "Inter", Segoe UI, Roboto, Helvetica, Arial, "Apple Color Emoji", "Segoe U
    letter-spacing: .1px;
  }

  /* Headings */
  h1,h2,h3 {
    color: var(--text);
    margin: 0 0 .5rem 0;
  }
  h1 { font-weight: 800; font-size: 2.2rem; letter-spacing: -0.02em; }
  h2 { font-weight: 700; font-size: 1.25rem; color: var(--muted); }

  /* ====== Sidebar ====== */
  [data-testid="stSidebar"] {
    background: var(--surface) !important;
    border-right: 1px solid #0b0d11;
  }
  [data-testid="stSidebar"] * {
    color: var(--text) !important;
  }

  /* Selectbox chip text contrast */
  .stSelectbox [data-baseweb="select"] {
    color: var(--text);
  }
```

Figure 39 Code for theme

The dark theme is established by this CSS file, which also sets the font, borders, shadows, and surface and background colors. Additionally, the app injects it at runtime and styles metric cards and badges

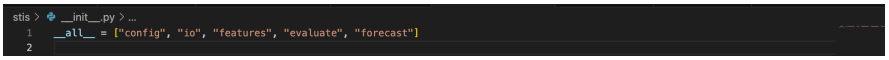


Figure 40 Code for stis-init.py

Figure 41 Code for models - init.py

Clean intra-package imports are made possible by this file, which makes the repository importable as a package. It facilitates tooling and modularity, guaranteeing consistent component reuse throughout the project.

Figure 42 Code for random forest singapore

Figure 43 Code for ridge regression bangkok

Figure 44 Code for seasonal naïve hong kong

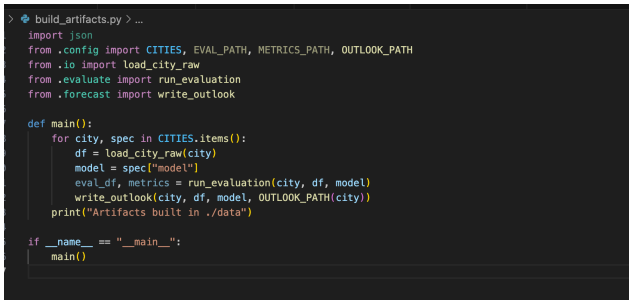
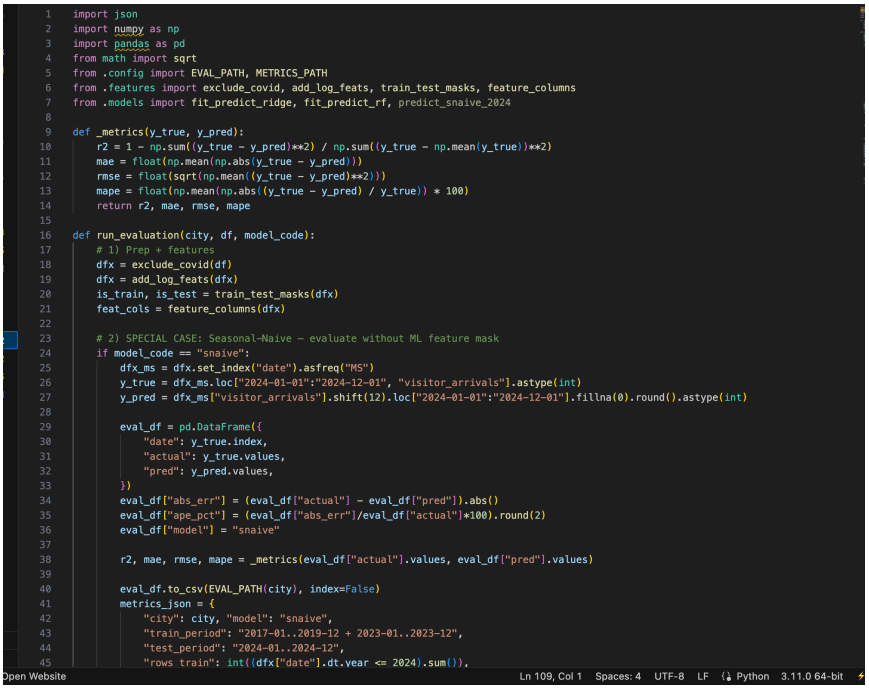


Figure 45 Code for building artefacts for streamlit UI

The batch driver is this script. All cities are cycled through, raw data is loaded, features are engineered, models are assessed, metrics are calculated, and 12-month outlooks are generated. It creates a single reproducible command that contains all the artifacts the user interface needs.

Figure 46 Code for configuration (Picking the best model for each city)



```python
import json
import numpy as np
import pandas as pd
from math import sqrt
from .config import EVAL_PATH, METRICS_PATH
from .features import exclude_covid, add_log_feats, train_test_masks, feature_columns
from .models import fit_predict_ridge, fit_predict_rf, predict_snaive_2024

def _metrics(y_true, y_pred):
    r2 = 1 - np.sum((y_true - y_pred)**2) / np.sum((y_true - np.mean(y_true))**2)
    mae = float(np.mean(np.abs(y_true - y_pred)))
    rmse = float(sqrt(np.mean((y_true - y_pred)**2)))
    mape = float(np.mean(np.abs((y_true - y_pred) / y_true)) * 100)
    return r2, mae, rmse, mape

def run_evaluation(city, df, model_code):
    # 1) Prep + features
    dfx = exclude_covid(df)
    dfx = add_log_feats(dfx)
    is_train, is_test = train_test_masks(dfx)
    feat_cols = feature_columns(dfx)

    # 2) SPECIAL CASE: Seasonal-Naive — evaluate without ML feature mask
    if model_code == "snaive":
        dfx_ms = dfx.set_index("date").asfreq("MS")
        y_true = dfx_ms.loc["2024-01-01":"2024-12-01", "visitor_arrivals"].astype(int)
        y_pred = dfx_ms["visitor_arrivals"].shift(12).loc["2024-01-01":"2024-12-01"].fillna(0).round().astype(int)

        eval_df = pd.DataFrame({
            "date": y_true.index,
            "actual": y_true.values,
            "pred": y_pred.values,
        })
        eval_df["abs_err"] = (eval_df["actual"] - eval_df["pred"]).abs()
        eval_df["ape_pct"] = (eval_df["abs_err"]/eval_df["actual"]*100).round(2)
        eval_df["model"] = "snaive"

        r2, mae, rmse, mape = _metrics(eval_df["actual"].values, eval_df["pred"].values)

        eval_df.to_csv(EVAL_PATH(city), index=False)
        metrics_json = {
            "city": city, "model": "snaive",
            "train_period": "2017-01..2019-12 + 2023-01..2023-12",
            "test_period": "2024-01..2024-12",
            "rows_train": int((dfx["date"].dt.year <= 2024).sum()),
```

Figure 47 Code Snippet for evaluation of the models

Fit, forecast, and score the 2024 hold-out with evaluate.py

This module trains on the engineered training set, selects models based on city, and performs strictly out-of-sample evaluations on 2024. Seasonal-Naïve replicates the values from the prior year; Random Forest and XGBoost fit tree ensembles; Ridge employs scaled predictors and an L2 penalty. Levels are back-transformed from log predictions. Both evaluation CSVs and metrics/model-card JSONs are written, and metrics (MAE, RMSE, MAPE, and R2) are calculated.

```python
def add_log_feats(df: pd.DataFrame) -> pd.DataFrame:
    # Month dummies (drop-first)
    out["month"] = out["date"].dt.month
    dums = pd.get_dummies(out["month"], prefix="m", drop_first=True)
    out = pd.concat([out, dums], axis=1)
    return out

def train_test_masks(df: pd.DataFrame):
    y = df["date"].dt.year
    is_train = (y.between(2017, 2019)) | (y == 2023)
    is_test  = (y == 2024)
    return is_train, is_test

def feature_columns(df: pd.DataFrame):
    return [c for c in df.columns if c.startswith(("log_lag", "log_roll", "m_"))]
```

Figure 48 Code for features

The preprocessing policy is formalized in this module. It calculates rolling means (3, 6), log-transformed targets, lagged values (1, 3, 6, 12), monthly seasonal dummies, and a COVID dummy.

Additionally, time-aware masks for train (2017–2023) and test (2024) are defined, and the ordered list of feature columns for models is provided.

```
def outlook_next12(city, df, model_code):
        model = _fit_rf_valid(dfx, feat_cols)
        out = _recursive_forecast_log_model(dfx, feat_cols, model)

    elif model_code == "snaive":
        # Copy 2024 month-by-month into 2025
        dfx_ms = dfx.set_index("date").asfreq("MS")
        last_year = dfx_ms.loc["2024-01-01":"2024-12-01", "visitor_arrivals"]
        if last_year.isna().any():
            raise ValueError("SNaive requires complete 2024 to copy seasonality.")
        future_idx = pd.date_range("2025-01-01", periods=12, freq="MS")
        out = pd.DataFrame({"date": future_idx, "forecast": last_year.values})

    else:
        raise ValueError("Unknown model code")

    out["model"] = model_code
    out["notes"] = "Scenario forecast; COVID years excluded in training"
    return out

def write_outlook(city, df, model_code, path):
    out = outlook_next12(city, df, model_code)
    out.to_csv(path, index=False)
    return out
```

Figure 49 Code Snippet for forecasting

The 12-month outlook (scenario generation) is provided by forecast.py. A logical 12-month scenario is produced by this module. It recursively predicts forward and back-transforming to levels for log models by re-fitting historical data through 2024. It replicates the value from the prior year for Seasonal-Naïve. A neat monthly forecast series is the end result.

```
import pandas as pd
from .config import CITIES

REQUIRED = ["date", "visitor_arrivals"]

def load_city_raw(city: str) -> pd.DataFrame:
    path = CITIES[city]["raw"]
    df = pd.read_csv(path)

    # Standardize HK duplicate year/month columns if present
    for cand in ("year_x", "year_y"):
        if cand in df.columns: df = df.rename(columns={cand: "year"})
    for cand in ("month_x", "month_y"):
        if cand in df.columns: df = df.rename(columns={cand: "month"})

    # Date handling
    if "date" not in df.columns:
        raise ValueError(f"{city}: missing required 'date' column.")
    df["date"] = pd.to_datetime(df["date"], errors="raise")
    df = df.sort_values("date").drop_duplicates("date")
    df = df.set_index("date").asfreq("MS").reset_index()

    # Target
    if "visitor_arrivals" not in df.columns:
        raise ValueError(f"{city}: missing required 'visitor_arrivals' column.")
    df["visitor_arrivals"] = pd.to_numeric(df["visitor_arrivals"], errors="coerce")
    if df["visitor_arrivals"].isna().any():
        raise ValueError(f"{city}: NA found in visitor_arrivals after coercion.")

    # Optional exogenous → forward/back fill
    for exo in ("hotel_occupancy", "google_trends"):
        if exo in df.columns:
            df[exo] = pd.to_numeric(df[exo], errors="coerce").ffill().bfill()

    keep = [c for c in ["date","year","month","visitor_arrivals","hotel_occupancy","google_trends"] if c in df.columns]
    return df[keep]
```

Figure 50 Code for io.py

All reading and writing to disk is contained in this module. Loader functions enforce schema checks, coerce numeric types, create a monthly DateIndex, and read a city's raw CSV.
If required, optional auxiliary series can be safely forward- or back-filled. Writer functions include 12-month outlooks, metrics and model cards, and evaluation tables. The rest of the stack can assume clear, consistent data contracts by focusing I/O here.

5.5.2 Frontend

The layout of the interface is the same in every city. The user chooses a city (Hong Kong, Singapore, or Bangkok) from the left-hand sidebar. To ensure transparency about what the system is and is not intended to do, a note makes it clear that all forecasts are precomputed artifacts and that the 12-month outlook is a scenario rather than an evaluated forecast.

Each city's page in the main view starts with its name and the badge representing the final validated model (e.g., Seasonal-Naïve for Hong Kong, Random Forest for Singapore, and Ridge regression for Bangkok). The dashboard displays headline metrics in card format directly below: the model name, 2024 MAPE, 2024 RMSE, and a confidence indicator represented by a star rating.

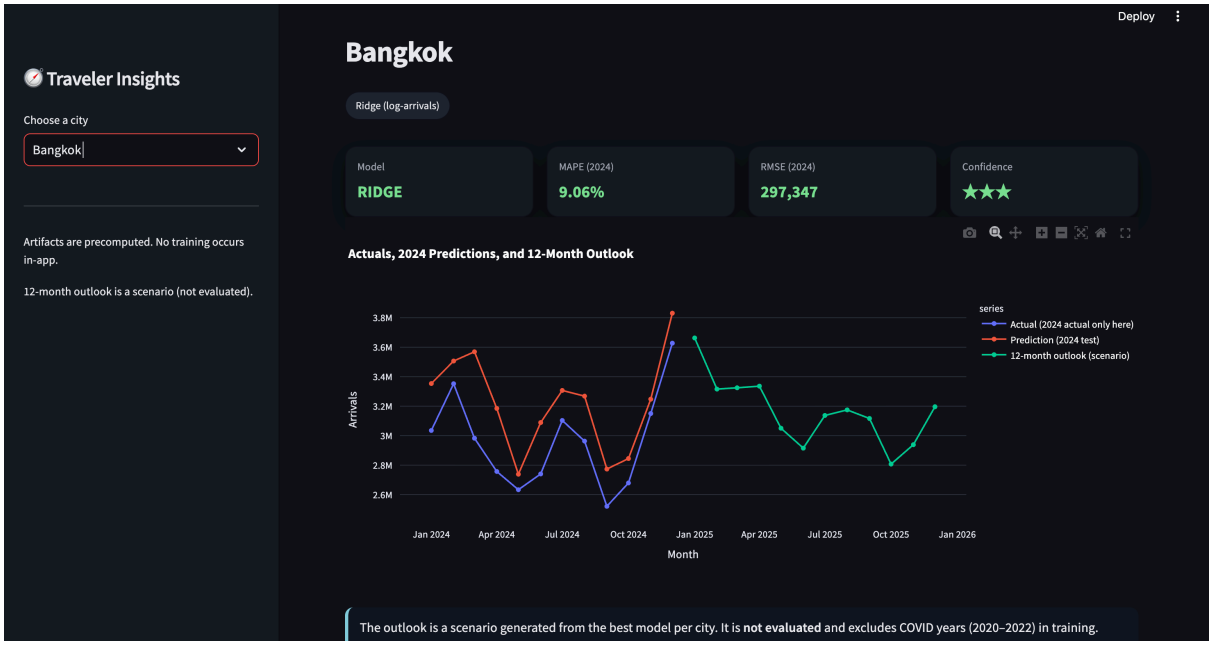


Figure 51 Bangkok's forecast

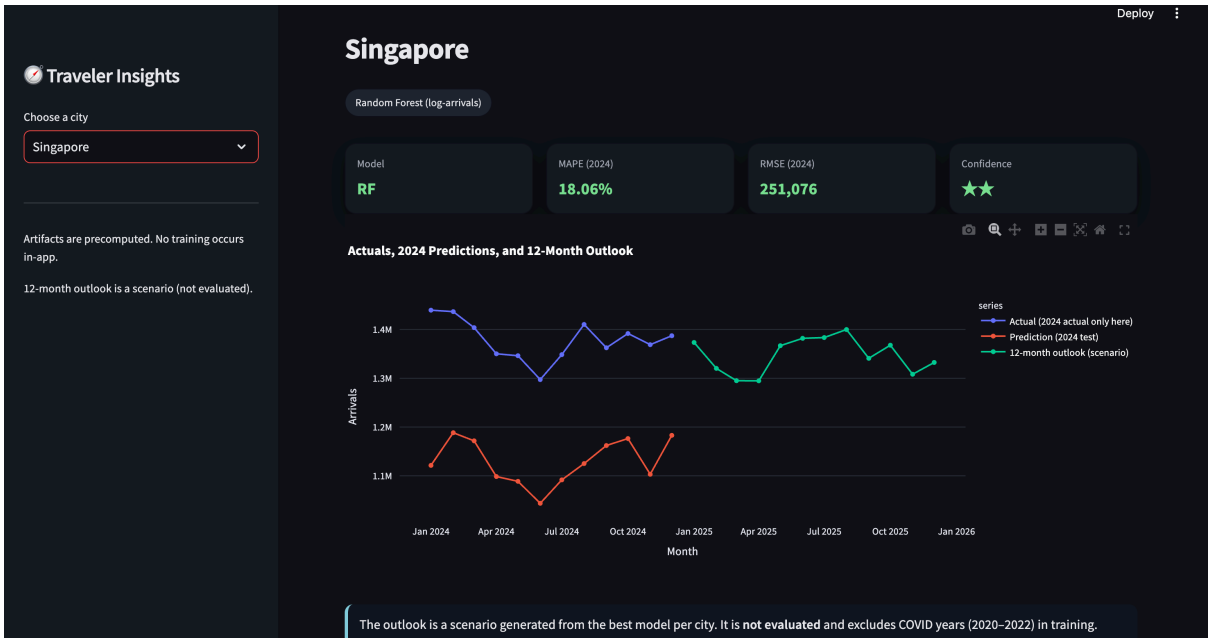


Figure 52 Singapore's Forecast

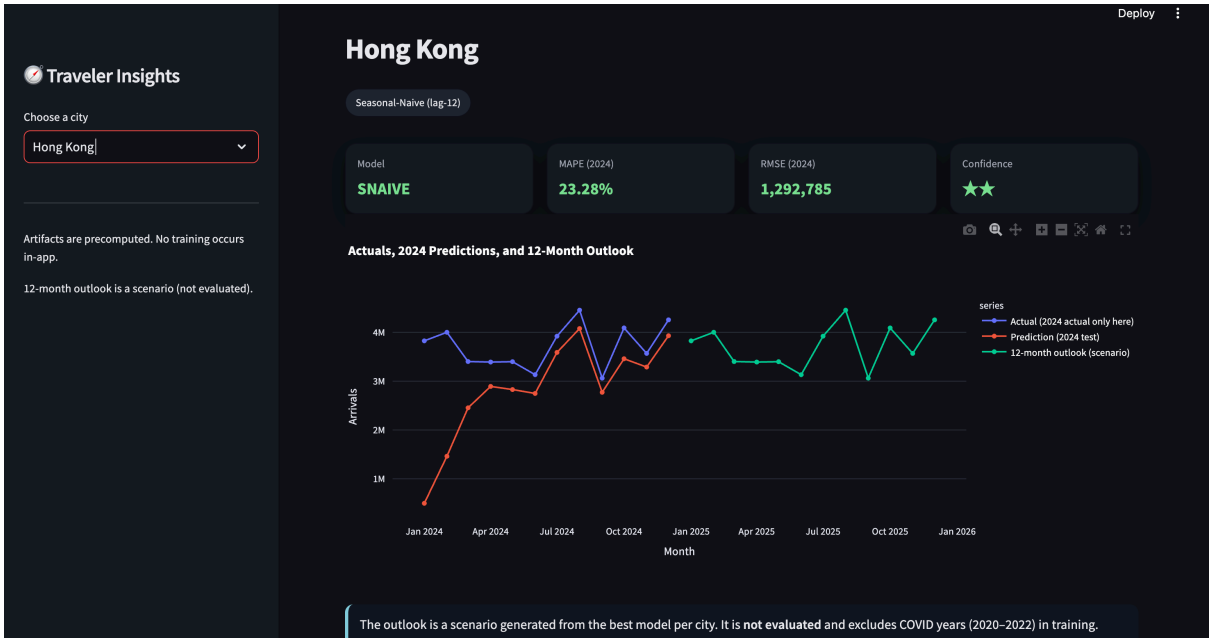
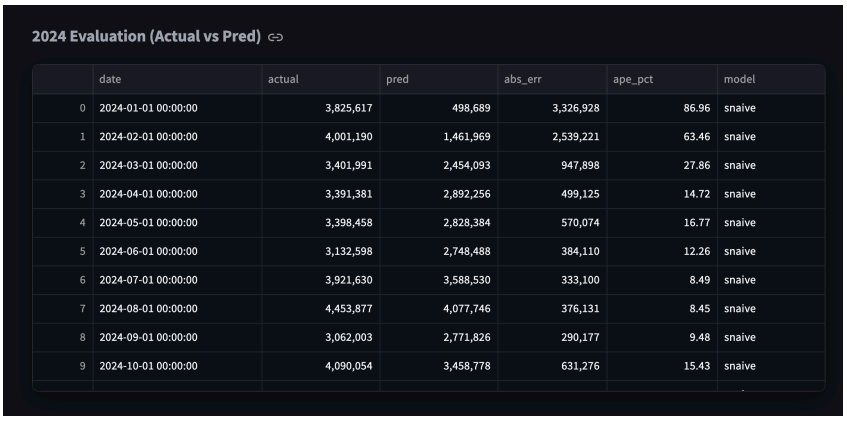


Figure 53 Hong Kong's forecast

An interactive Plotly time-series chart serves as the interface's focal point. Three important series are superimposed here: the 12-month scenario outlook (green), model predictions for the 2024 test set (red), and actual arrivals in 2024 (blue). It is simple to compare observed data with model predictions and see anticipated trends that will last until 2025 thanks to this

layered visualisation.

The methodological disclaimer that the outlook is scenario-based and does not include COVID years (2020–2022) in training is finally reaffirmed in a footer note. This guarantees that users are aware of the forecasts' origin and extent.

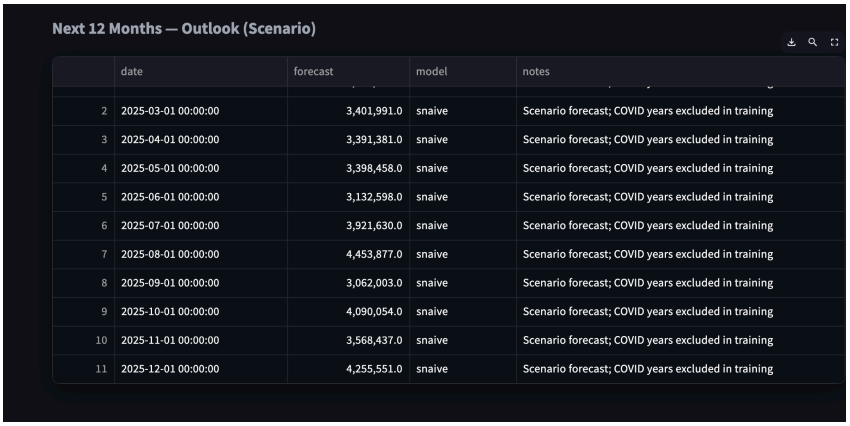


2024 Evaluation (Actual vs Pred)

	date	actual	pred	abs_err	ape_pct	model
0	2024-01-01 00:00:00	3,825,617	498,689	3,326,928	86.96	snaive
1	2024-02-01 00:00:00	4,001,190	1,461,969	2,539,221	63.46	snaive
2	2024-03-01 00:00:00	3,401,991	2,454,093	947,898	27.86	snaive
3	2024-04-01 00:00:00	3,391,381	2,892,256	499,125	14.72	snaive
4	2024-05-01 00:00:00	3,398,458	2,828,384	570,074	16.77	snaive
5	2024-06-01 00:00:00	3,132,598	2,748,488	384,110	12.26	snaive
6	2024-07-01 00:00:00	3,921,630	3,588,530	333,100	8.49	snaive
7	2024-08-01 00:00:00	4,453,877	4,077,746	376,131	8.45	snaive
8	2024-09-01 00:00:00	3,062,003	2,771,826	290,177	9.48	snaive
9	2024-10-01 00:00:00	4,090,054	3,458,778	631,276	15.43	snaive

Figure 54 Evaluations Actual vs prediction data

The 2024 Evaluation Table, which displays actual monthly arrivals along with model predictions, absolute error, and percentage error, is seen in this screenshot. Examiners can confirm precise values and error magnitudes monthly thanks to this tabular output, which adds a numerical complement to the time-series chart.



Next 12 Months — Outlook (Scenario)

	date	forecast	model	notes
2	2025-03-01 00:00:00	3,401,991.0	snaive	Scenario forecast; COVID years excluded in training
3	2025-04-01 00:00:00	3,391,381.0	snaive	Scenario forecast; COVID years excluded in training
4	2025-05-01 00:00:00	3,398,458.0	snaive	Scenario forecast; COVID years excluded in training
5	2025-06-01 00:00:00	3,132,598.0	snaive	Scenario forecast; COVID years excluded in training
6	2025-07-01 00:00:00	3,921,630.0	snaive	Scenario forecast; COVID years excluded in training
7	2025-08-01 00:00:00	4,453,877.0	snaive	Scenario forecast; COVID years excluded in training
8	2025-09-01 00:00:00	3,062,003.0	snaive	Scenario forecast; COVID years excluded in training
9	2025-10-01 00:00:00	4,090,054.0	snaive	Scenario forecast; COVID years excluded in training
10	2025-11-01 00:00:00	3,568,437.0	snaive	Scenario forecast; COVID years excluded in training
11	2025-12-01 00:00:00	4,255,551.0	snaive	Scenario forecast; COVID years excluded in training

Figure 55 Outlook for the next 12 months

The Outlook Table for the Next 12 Months is the subject of this screenshot. In this case, the system produces predicted values for every month in 2025, each one prominently labeled with the model that was employed and accompanied by annotations that make clear the origin

of the scenario (e.g., that COVID years were excluded from training). This strengthens the forecasts' methodological integrity and reproducibility.

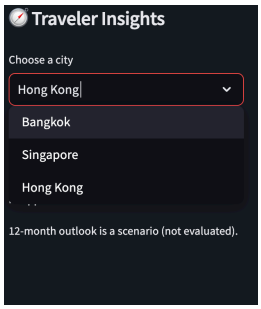


Figure 56 Drop down bar

The City Selector dropdown in the sidebar is shown in the screenshot. It guarantees a uniform workflow and lists the three supported cities: Bangkok, Singapore, and Hong Kong. The metrics, charts, tables, and downloads related to a destination are dynamically refreshed when a city is selected.

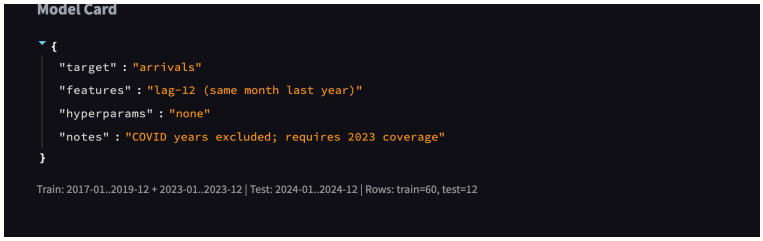


Figure 57 Model Card

A clear synopsis of the model specification and provenance for the chosen city can be found in the Model Card section of this interface. The target variable, features used, hyperparameters (if any), and any methodological notes are all detailed in a structured JSON object. The target in the Hong Kong example is arrivals, the feature is lag-12 (the same month's arrivals from the prior year, in line with the Seasonal-Naïve approach), and the hyperparameters are marked as "none," indicating that this baseline method lacks tunable parameters. The notes point out that COVID years (2020–2022) were not included in the training, and that in order to enable reliable year-over-year comparisons, 2023 coverage had to be kept.

5.5.3 Summary

A transparent frontend and a modular backend, both created to meet academic standards for reproducibility and interpretability, enabled the Smart Traveller Insights system's deployment. An offline–online pipeline was used to implement the backend. Prior to feature

engineering, which included log-transformations, lag variables, rolling means, seasonal dummies, and COVID-period adjustments, raw data were ingested, validated, and standardized. Following rigorous training and evaluation of the models on a hold-out test set, the results were recorded in structured artifacts that included evaluation tables, error metrics, model cards, and scenarios for the future. Because this architecture kept computation and presentation separate, all training could be done offline and the results could still be audited and replicated.

As a serve-only dashboard, the frontend was created in Streamlit to consume offline-generated artifacts and display them in an intuitive manner. The validated model badge, key performance indicators, and a confidence rating were displayed in each view as users navigated between cities using a sidebar selector. Actuals, forecasts, and scenario outlooks were superimposed on interactive time-series charts, and accurate error breakdowns and forecast values were presented in tabular views. Model cards recorded target variables, feature sets, hyperparameters, and training/testing splits, while download buttons provided direct access to evaluation and outlook datasets. To guarantee uniformity, readability, and a polished appearance, the interface was designed using a unique theme and reusable elements.

The frontend and backend work together to show a deployment strategy that is both practically interpretable and rigorously methodological. The frontend converts these outputs into an approachable decision-support tool that blends accuracy, transparency, and usability, while the backend ensures repeatable, thoroughly documented forecasts.

CHAPTER 6: CONCLUSION

6.1 Critical Evaluation

The goal of this study was to create the Smart Traveller Insights system as a forecasting and decision-support tool for Asia-Pacific's tourism recovery after the pandemic. By concentrating on three major locations—Hong Kong, Singapore, and Bangkok—the project showed how data-driven modelling can capture the disruptions brought on by structural shocks like COVID-19 as well as the continuity of seasonal patterns.

Model suitability is highly context-dependent, according to the research. The best method for Bangkok turned out to be ridge regression, which made use of strong predictor correlations to produce incredibly accurate forecasts. Random Forest was verified as the final model for Singapore, offering resilience in low-variance scenarios where variance-based measures like R2 were less significant. The Seasonal-Naïve model was the most convincing in Hong Kong, highlighting the significance of clear, uncomplicated baselines in unstable recovery settings. All of these results support the idea that forecasting frameworks need to be flexible, with model selection based on the data's structural features rather than just algorithmic complexity.

The study emphasized the significance of strong validation procedures from a methodological standpoint. Error diagnostics, benchmark comparisons, and chronological splits made sure that the models were both accurate and generalizable. The limitations of any one measure were lessened by the purposeful use of multiple metrics, especially in low variance contexts. The study acknowledged the circumstances in which more straightforward approaches can still be competitive while showcasing the value added by sophisticated approaches by incorporating naive baselines.

The project produced a fully functional deployment pipeline in addition to accurate forecasts. In order to create reproducible artifacts, the backend modularized the data ingestion, feature engineering, evaluation, and outlook generation processes. These artifacts were converted by the frontend into a serve-only Streamlit dashboard that gave users access to model cards, tabular evidence, interactive visualizations, key performance indicators, and downloadable datasets. This design demonstrated how research outputs can be operationalized into interpretable, decision-support tools by fusing academic rigor with practical usability.

In conclusion, the Smart Traveller Insights system serves as an example of how, in the wake of unprecedented disruption, tourism forecasting can be rethought. By demonstrating that transparent, validated, and context-sensitive models can produce useful insights even in unstable recovery situations, it advances the body of literature. It also highlights the importance of integrating forecasting into easily accessible dashboards and reproducible pipelines, which connects methodological depth with practical impact. Because of these contributions, the system is positioned as a useful prototype for aiding in tourism recovery planning as well as a scholarly exercise in methodological innovation.

6.1 Recommendations and Future Research

Although the Smart Traveller Insights system has proven to be capable of offering trustworthy predictions and clear decision-support, a number of improvements could improve both its methodological soundness and its usefulness in real-world scenarios. Opportunities to increase coverage, enhance modelling depth, and widen application are covered by these suggestions.

First, future studies ought to think about extending the system's geographic reach. Three representative cities—Bangkok, Singapore, and Hong Kong—were chosen for the current study due to their regional importance and the availability of reliable monthly data. Comparative insights into diverse recovery trajectories would be obtained by expanding the framework to include more locations throughout Southeast Asia and the larger Asia-Pacific region. Comparing established and developing tourism markets could be made easier by incorporating multi-country datasets.

Second, more diverse exogenous variables might be advantageous for the system. Historical arrivals, hotel occupancy rates, and Google Trends indices were the main sources of data used in the current models. Future versions might incorporate environmental factors (air quality indices, extreme weather events), policy-related variables (visa restrictions, travel advisories), and macroeconomic indicators (exchange rates, fuel prices, income indices). Such predictors would increase the model's sensitivity to outside shocks and bring it closer to the intricate factors influencing demand for travel.

Third, advanced modelling techniques that strike a balance between interpretability and complexity should be investigated in future research. Performance in unstable environments may be improved by hybrid approaches, such as ensemble systems that dynamically weight forecasts or combining machine learning with conventional econometric models. If enough data volume and quality are available, deep learning architectures like temporal convolutional networks and LSTMs could be tested. It is crucial to maintain the interpretability of these approaches, either by using transparent benchmark models or by employing post-hoc explainability techniques.

6.2 Limitations

The Smart Traveller Insights system has a number of drawbacks that should be noted despite its advantages. These limitations set the parameters for how the results should be interpreted and are mostly related to the model's scope, data availability, and methodological presumptions.

First, the consistency and accessibility of data across cities limited the analysis. Other possible locations were disqualified because of missing or inconsistent time series, even though Bangkok, Singapore, and Hong Kong were chosen because of their comparatively strong datasets. Auxiliary variables like hotel occupancy and Google Trends required imputation or proxy measures in some locations, even within the chosen cities, which could have introduced bias.

Second, the study only looked at a small number of predictors, mostly Google Trends indices, hotel occupancy, and historical arrivals. The entire spectrum of macroeconomic, political, and environmental factors impacting recovery is not fully captured by these variables, despite the fact that they are significant indicators of tourism demand. The models' explanatory power may have been limited by excluding factors like exchange rates, travel restrictions, and global fuel prices, especially during times of structural volatility.

Third, a twelve-month scenario outlook was included, but the forecasting horizon was restricted to one year out-of-sample. Although methodological stability was guaranteed by this short horizon, forecasts' applicability for long-term strategic planning may be limited.

References

- Adekuajo, I. O., Otokiti, B. O., & Okpeke, F. (2025). A predictive framework for post-pandemic tourism recovery: Integrating machine learning and visitor behavior analytics. *SHISRRJ.* https://shisrrj.com/paper/SHISRRJ258315.pdf
- Afrianto, M. A., & Wasesa, M. (2022). The impact of tree-based machine learning models, length of training data, and quarantine search query on tourist arrival prediction's accuracy under COVID-19. *Current Issues in Tourism.* https://doi.org/10.1080/13683500.2022.2085079
- Akter, S., Bandara, R., & Hossain, M. N. (2023). Dashboard technologies for decision support in tourism recovery. *Tourism Management Perspectives, 46,* 101085. https://doi.org/10.1016/j.tmp.2023.101085
- Assaf, A. G., & Li, G. (2022). Tourism forecasting post-COVID-19: Recovery patterns and methodological implications. *Tourism Management, 90,* 104485. https://doi.org/10.1016/j.tourman.2021.104485
- Assaf, A. G., Song, H., & Wu, D. C. (2022). Big data applications in tourism demand forecasting: A review. *Annals of Tourism Research, 92,* 103327. https://doi.org/10.1016/j.annals.2022.103327
- Bi, J. W., Han, T. Y., Yao, Y., & Yang, T. (2024). Tourism demand forecasting under conceptual drift during COVID-19: An ensemble deep learning model. *Current Issues in Tourism.* https://doi.org/10.1080/13683500.2023.2273922
- Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32. https://doi.org/10.1023/A:1010933404324
- Chen, H., Wang, Y., & Li, X. (2022). Machine learning models for tourism demand forecasting: Advances and challenges. *Annals of Tourism Research, 92,* 103310. https://doi.org/10.1016/j.annals.2021.103310
- Chen, H., Wang, Y., & Li, X. (2023). Tourism demand forecasting during and after COVID-19: Outliers and recovery scenarios. *Journal of Travel Research, 62*(5), 855–870. https://doi.org/10.1177/00472875221150604
- Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). https://doi.org/10.1145/2939672.2939785

- Chen, Y., Hu, T., & Law, R. (2025). Tourism demand forecasting: A novel multi-channel imaging model. *Journal of Tourism Futures.* https://doi.org/10.1108/jtf-12-2024-0263
- Choi, H., & Varian, H. (2012). Predicting the present with Google Trends. *Economic Record, 88*(s1), 2–9. https://doi.org/10.1111/j.1475-4932.2012.00809.x
- Dimitriadou, A., & Gogas, P. (2025). Tourism and uncertainty: A machine learning approach. *Current Issues in Tourism.* https://doi.org/10.1080/13683500.2024.2370380
- Goh, C., & Law, R. (2002). Modelling and forecasting tourism demand for arrivals with stochastic nonstationary seasonality and intervention. *Tourism Management, 23*(5), 499–510. https://doi.org/10.1016/S0261-5177(02)00009-2
- Gretzel, U., Werthner, H., Koo, C., & Lamsfus, C. (2020). Conceptual foundations for understanding smart tourism ecosystems. *Computers in Human Behavior, 110,* 106340. https://doi.org/10.1016/j.chb.2019.106340
- Gunter, U., & Önder, I. (2021). Forecasting city tourism: Machine learning versus structural models. *Tourism Economics, 27*(3), 531–550. https://doi.org/10.1177/1354816620926391
- Gunter, U., & Önder, I. (2023). Forecasting tourism demand with machine learning models: Recent developments and future directions. *Tourism Economics, 29*(2), 305–327. https://doi.org/10.1177/13548166221101234
- Hall, C. M. (2022). Tourism and pandemics: Towards a resilient future. *Tourism Review International, 26*(2), 117–133. https://doi.org/10.3727/154427222X16490830883520
- Hall, C. M., Prayag, G., & Amore, A. (2021). Tourism and resilience in the face of crisis. *Journal of Sustainable Tourism, 29*(9), 1421–1437. https://doi.org/10.1080/09669582.2021.1910213
- Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics, 12*(1), 55–67. https://doi.org/10.1080/00401706.1970.10488634
- Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and practice* (2nd ed.). OTexts. https://otexts.com/fpp2/
- Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and practice* (3rd ed.). OTexts. https://otexts.com/fpp3/

- Kayral, İ. E., Sarı, T., & Tandoğan Aktepe, N. Ş. (2023). Forecasting tourist arrivals and income with combined ANN architecture in the post-COVID-19 period: The case of Turkey. *Sustainability, 15*(22), 15924. https://doi.org/10.3390/su152215924
- Kuhn, M., & Johnson, K. (2020). *Feature engineering and selection: A practical approach for predictive models.* CRC Press.
- Kumar, A., Misra, S. C., & Chan, F. T. S. (2022). Leveraging AI for advanced analytics to forecast altered tourism industry parameters: A COVID-19 motivated study. *Expert Systems with Applications, 206,* 117736. https://doi.org/10.1016/j.eswa.2022.117736
- Lee, G. C. (2025). A data-driven approach to tourism demand forecasting: Integrating web search data into a SARIMAX model. *Data, 10*(5), 73. https://doi.org/10.3390/data10050073
- Li, G., Song, H., & Witt, S. F. (2005). Recent developments in econometric modelling and forecasting. *Journal of Travel Research, 44*(1), 82–99. https://doi.org/10.1177/0047287505276594
- Li, G., Song, H., & Guo, Z. (2022). Evaluating forecasting accuracy in tourism demand modelling. *Annals of Tourism Research, 95,* 103398. https://doi.org/10.1016/j.annals.2022.103398
- Li, G., Song, H., & Witt, S. F. (2022). Recent developments in international tourism demand forecasting: Coping with shocks and structural breaks. *International Journal of Forecasting, 38*(2), 429–444. https://doi.org/10.1016/j.ijforecast.2021.09.005
- Li, Y., Yang, D., Guo, J., Sun, S., & Wang, S. (2024). Daily tourism demand forecasting before and during COVID-19: Data predictivity and an improved decomposition-ensemble framework. *Current Issues in Tourism, 27*(10), 1523–1542. https://doi.org/10.1080/13683500.2023.2202308
- Liu, Y., Xu, H., & Li, X. (2023). A composite tourism recovery index for Southeast Asia: Integrating mobility, hospitality, and aviation data. *Tourism Economics, 29*(6), 1362–1381. https://doi.org/10.1177/13548166231154723
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020). The M4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting, 36*(1), 54–74. https://doi.org/10.1016/j.ijforecast.2019.04.014
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020). The M5 competition: Findings and implications. *International Journal of Forecasting, 36*(1), 54–74. https://doi.org/10.1016/j.ijforecast.2019.07.001

- Moreno-Izquierdo, L., Más-Ferrando, A., & Ribalaygua, C. (2024). Evaluating machine learning techniques for predicting tourist occupancy: An experiment with pre- and post-pandemic COVID-19 data. *Current Issues in Tourism.* https://doi.org/10.1080/13683500.2023.2282163
- Nguyen, D. T., Li, Y., Peng, C. L., & Cho, M. Y. (2024). Monthly tourism demand forecasting with COVID-19 impact-based hybrid convolutional neural network and gate recurrent unit. *International Journal of Tourism Research.* https://doi.org/10.1002/jtr.2812
- Nguyen, T. H., Pham, H. T., & Tran, M. T. (2022). Building a tourism recovery dashboard: A case study from Vietnam. *Journal of Destination Marketing & Management, 26,* 100700. https://doi.org/10.1016/j.jdmm.2022.100700
- Polyzos, S., Fotiadis, A., & Samitas, A. (2021). COVID-19 tourism recovery in the ASEAN and East Asia region: Asymmetric patterns and implications. *ERIA Discussion Paper Series.* https://www.eria.org/publications/covid-19-tourism-recovery-in-the-asean-and-east-asia-region-asymmetric-patterns-and-implications
- Qiu, R. T. R., Wu, D. C., Dropsy, V., Petit, S., & Pratt, S. (2021). Visitor arrivals forecasts amid COVID-19: A perspective from the Asia and Pacific team. *Annals of Tourism Research, 88,* 103155. https://doi.org/10.1016/j.annals.2020.103155
- Qiu, R. T. R., Wu, D. C., Dropsy, V., Petit, S., & Pratt, S. (2021). Tourism forecasting amid COVID-19: The case of Macao. *Annals of Tourism Research, 88,* 103170. https://doi.org/10.1016/j.annals.2021.103170
- Saltz, J. (2021). Data science project management: Toward a comprehensive approach. *Journal of Information Systems, 35*(1), 131–144. https://doi.org/10.2308/isys-19-047
- Shang, Y., Wang, Y., & Chan, C. (2021). Measuring model performance in low-variance time series: A case in tourism demand. *Tourism Economics, 27*(8), 1744–1762. https://doi.org/10.1177/13548166211004165
- Chumbar, S. (2023, September 24). *Knowledge Discovery in Databases (KDD): A Practical Approach*. Medium. https://medium.com/@shawn.chumbar/knowledge-discovery-in-databases-kdd-a-practical-approach-f28247493be4
- Shahrabi, A., Rajabifard, A., & Kalantari, M. (2022). Designing decision support systems for resilient urban tourism under COVID-19. *Sustainability, 14*(18), 11632. https://doi.org/10.3390/su141811632

- Shimaoka, A. M., Ferreira, R. C., & Oliveira, R. C. (2024). The evolution of CRISP-DM for data science: Methods, processes, and frameworks. *SBC Journal on Interactive Systems, 15*(1), 1–14. https://doi.org/10.5753/jis.2024.3128
- Song, H., & Li, G. (2008). Tourism demand modelling and forecasting – A review of recent research. *Tourism Management, 29*(2), 203–220. https://doi.org/10.1016/j.tourman.2007.07.016
- Song, H., & Li, G. (2021). Tourism forecasting: A review of recent research. *Tourism Economics, 27*(6), 1185–1205. https://doi.org/10.1177/1354816620978754
- Song, H., Qiu, R. T. R., & Park, J. (2019). A review of research on tourism demand forecasting: Launching the *Annals of Tourism Research* curated collection on tourism demand forecasting. *Annals of Tourism Research, 75,* 338–362. https://doi.org/10.1016/j.annals.2019.01.004
- Song, H., Qiu, R. T. R., & Park, J. (2021). Short-term forecasting of tourism demand under crisis conditions. *Annals of Tourism Research, 87,* 103143. https://doi.org/10.1016/j.annals.2020.103143
- Sun, S., Wei, Y., & Wu, D. (2021). Practical approaches to handling non-stationarity in tourism time series forecasting. *Tourism Economics, 27*(7), 1555–1575. https://doi.org/10.1177/1354816621997923
- Sulong, Z., & Abdullah, M. (2023). Halal tourism demand and firm performance forecasting: New evidence from machine learning. *Current Issues in Tourism.* https://doi.org/10.1080/13683500.2022.2145458
- Truong, V. D., Hall, C. M., & Garry, T. (2022). Digital innovation in tourism recovery strategies. *Journal of Destination Marketing & Management, 23,* 100699. https://doi.org/10.1016/j.jdmm.2022.100699
- UNWTO. (2022). Tourism recovery tracker. World Tourism Organization. https://www.unwto.org/tourism-data/unwto-tourism-recovery-tracker
- Vanneste, D. (2022). Tourism, uncertainty, and well-being: Rethinking resilience. *Tourism Geographies, 24*(5–6), 745–761. https://doi.org/10.1080/14616688.2022.2035824
- Velu, S. R., Ravi, V., & Tabianan, K. (2022). Predictive analytics of COVID-19 cases and tourist arrivals in ASEAN. *Health and Technology, 12*(6), 1325–1340. https://doi.org/10.1007/s12553-022-00701-7

- Witt, S. F., & Witt, C. A. (1995). Forecasting tourism demand: A review of empirical research. *International Journal of Forecasting, 11*(3), 447–475. https://doi.org/10.1016/0169-2070(95)00591-7
- Yang, D., Li, Y., Guo, J., & Sun, S. (2023). Regional tourism demand forecasting with spatiotemporal interactions: A multivariate decomposition deep learning model. *Asia Pacific Journal of Tourism Research, 28*(4), 405–421. https://doi.org/10.1080/10941665.2023.2256431
- Yang, Y., Li, G., Guo, Z., & Sun, J. (2023). Hybrid forecasting approaches for post-crisis tourism recovery: Evidence from international travel flows. *Journal of Travel Research, 62*(5), 909–926. https://doi.org/10.1177/00472875221110045
- Zhang, Y., Wang, H., & Liu, S. (2023). Feature engineering for time series forecasting: Advances and applications. *Applied Soft Computing, 132,* 109835. https://doi.org/10.1016/j.asoc.2022.109835
- Zheng, J., Ma, X., Wang, D., Li, P., & Yu, Y. (2025). Improved multi-step prediction of daily tourism demand: An innovative hybrid machine learning framework with search engine data. *Current Issues in Tourism.* https://doi.org/10.1080/13683500.2025.2554872

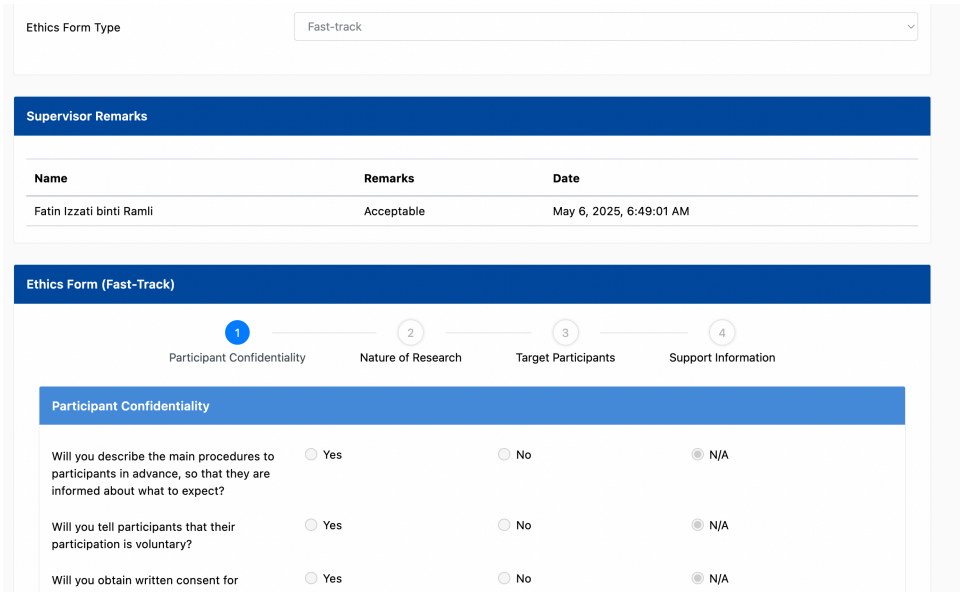
APPENDIX

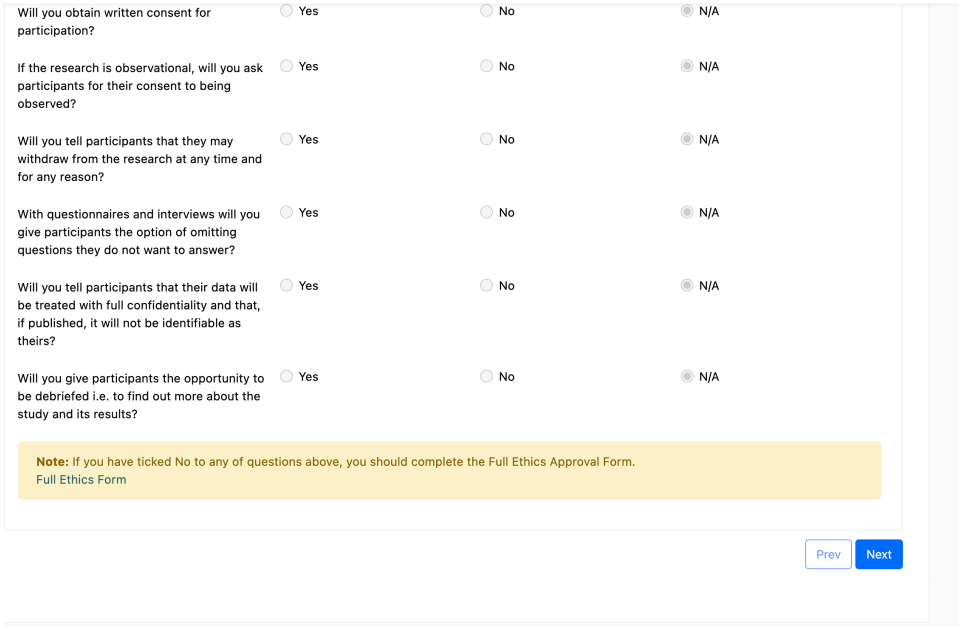
1. Project Proposal

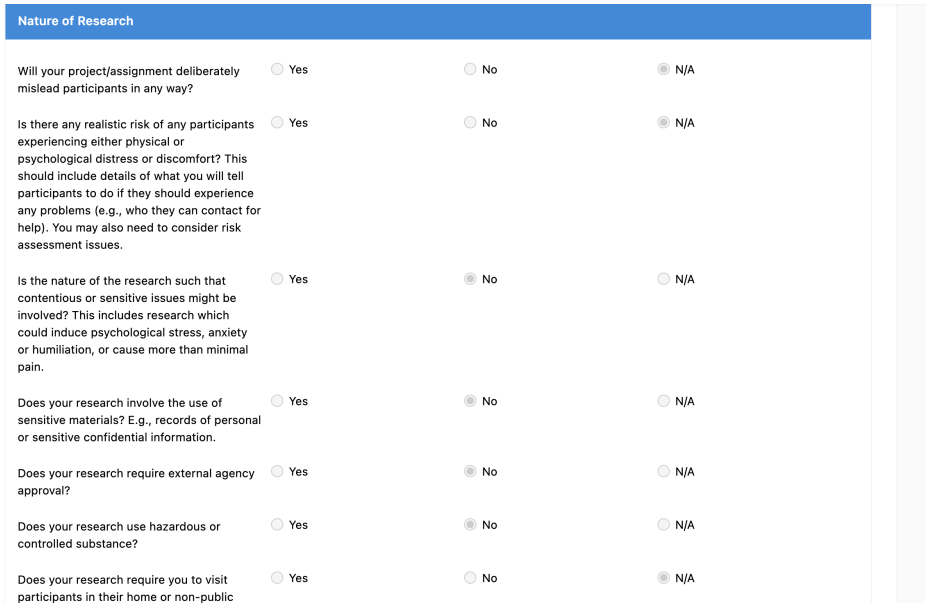
Project Title Proposal

Project Title: Developing a Smart Traveler Insights System to Analyse Post Pandemic Tourism Recovery in Asia Pacific		**Status**	APPROVED
Proposed By	ABDUL MUHAIMIN AMAN		
Description	1. Introduction Global tourism was hampered by the COVID-19 epidemic, leaving many tourists unsure of where and when to go safely. Even after five years, worldwide tourism has not yet recovered to its pre-pandemic levels, and many places are still having difficulty attracting the same number of tourists as before. While some cities in the Asia-Pacific area have recovered more quickly than others, others are still dealing with declining visitor confidence, economic losses, and shifting travel regulations. Travellers find it challenging to identify whether cities are secure, completely recovered, and worthwhile because to this discrepancy. Nowadays, the majority of travel choices are made based on general travel advisories, social media evaluations, and word-of-mouth—all of which frequently lack up-to-date accuracy. No unified system exists that offers unbiased, data-driven insights on the patterns of tourist recovery. This project aims to develop a Smart Traveler Insights System, a web platform that analyses real-time travel data, sentiment analysis, and tourism statistics to generate a Traveler Confidence Score for different cities. By categorising destinations into Luxury, Cultural, Nature & Eco-Tourism, and Urban & Business Tourism, the system helps travellers make informed decisions on where to visit based on safety, popularity, and recovery trends. 2. Aim To create a Smart Traveler Insights System that uses sentiment analysis, machine learning, and real-time travel data to assess and forecast tourism recovery patterns in Asia-Pacific cities, allowing tourism stakeholders to make better decisions.		
	2. Aim To create a Smart Traveler Insights System that uses sentiment analysis, machine learning, and real-time travel data to assess and forecast tourism recovery patterns in Asia-Pacific cities, allowing tourism stakeholders to make better decisions. 3. Objectives Determine the main elements influencing visitor confidence, such as safety, economic stability, and travel limitations, by analysing tourist recovery patterns in Asia-Pacific cities. Gather and analyse current travel information from sources such as government tourism reports, social media sentiment, UNWTO, and Google Trends. Create a web-based Smart Traveler Insights System that uses sentiment analysis and machine learning to create a Traveler Confidence Score and display tourism recovery trends. Using data patterns and passenger sentiment, enable predictive analytics to predict future tourist trends. Test accuracy, usability, and real-time data validation to gauge the system's efficacy. 4. Targeted Users Government Tourism Boards & Policymakers Hospitality & Travel Industry (Hotels, Airlines, Tour Operators) Travellers (Tourists & Digital Nomads) Researchers & Analysts in Tourism & Economic Development		
SDG	SDG3		
Keywords	Machine Learning Natural Language Processing (NLP)		Web-Based Application
Preferred Supervisor(s)	DR. MASRINA AKMAL BINTI SALLEH, AZIAH BINTI ABDOLLAH, ASSOC. PROF. DR. IMRAN MEDI, ASSOC. PROF. DR. SHAHRINAZ BINTI ISMAIL, DR. PREETHI SUBRAMANIAN		
Assigned Supervisor	fatin.ramli		
Assigned Second Marker	maryting		

2. Ethics Form

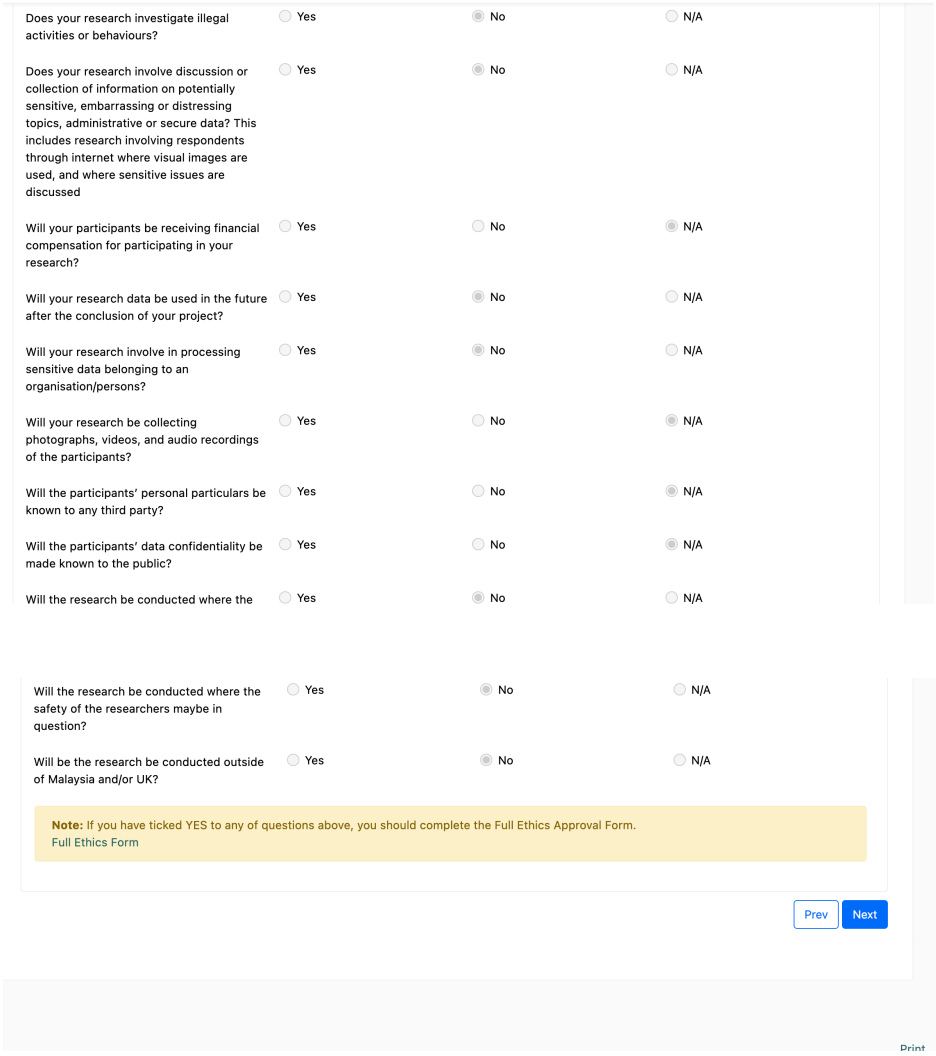






Nature of Research

Will your project/assignment deliberately mislead participants in any way?	Yes	No	N/A
Is there any realistic risk of any participants experiencing either physical or psychological distress or discomfort? This should include details of what you will tell participants to do if they should experience any problems (e.g., who they can contact for help). You may also need to consider risk assessment issues.	Yes	No	N/A
Is the nature of the research such that contentious or sensitive issues might be involved? This includes research which could induce psychological stress, anxiety or humiliation, or cause more than minimal pain.	Yes	No	N/A
Does your research involve the use of sensitive materials? E.g., records of personal or sensitive confidential information.	Yes	No	N/A
Does your research require external agency approval?	Yes	No	N/A
Does your research use hazardous or controlled substance?	Yes	No	N/A
Does your research require you to visit participants in their home or non-public	Yes	No	N/A



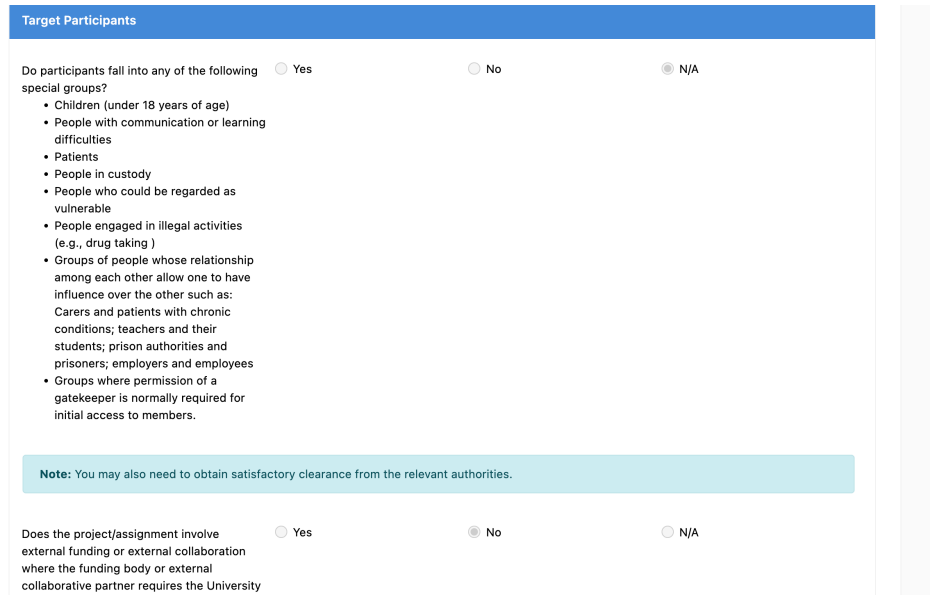
Does your research investigate illegal activities or behaviours?	Yes	No	N/A
Does your research involve discussion or collection of information on potentially sensitive, embarrassing or distressing topics, administrative or secure data? This includes research involving respondents through internet where visual images are used, and where sensitive issues are discussed	Yes	No	N/A
Will your participants be receiving financial compensation for participating in your research?	Yes	No	N/A
Will your research data be used in the future after the conclusion of your project?	Yes	No	N/A
Will your research involve in processing sensitive data belonging to an organisation/persons?	Yes	No	N/A
Will your research be collecting photographs, videos, and audio recordings of the participants?	Yes	No	N/A
Will the participants' personal particulars be known to any third party?	Yes	No	N/A
Will the participants' data confidentiality be made known to the public?	Yes	No	N/A
Will the research be conducted where the	Yes	No	N/A

Will the research be conducted where the safety of the researchers maybe in question?	Yes	No	N/A
Will be the research be conducted outside of Malaysia and/or UK?	Yes	No	N/A

Note: If you have ticked YES to any of questions above, you should complete the Full Ethics Approval Form.
Full Ethics Form

Prev Next

Print



Target Participants

Do participants fall into any of the following special groups?

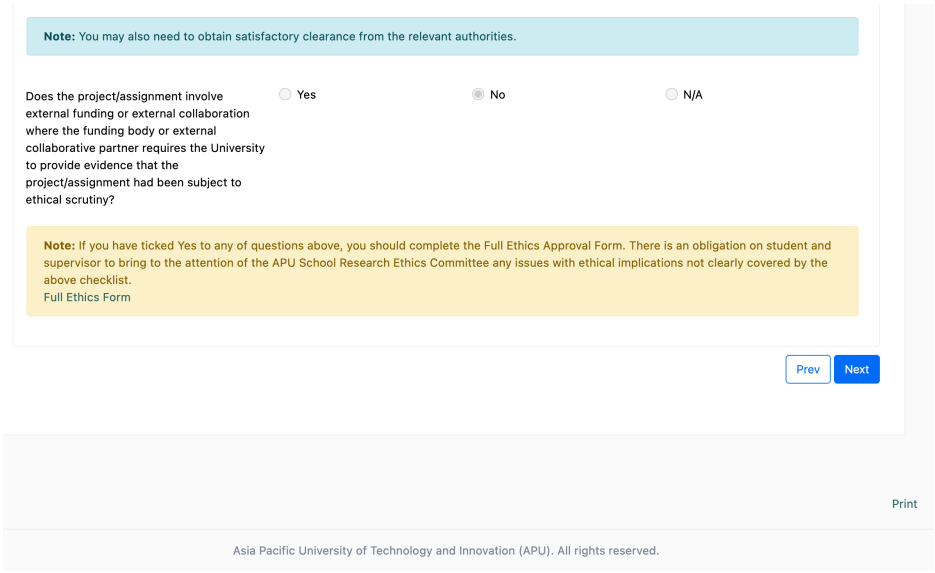
- Children (under 18 years of age)
- People with communication or learning difficulties
- Patients
- People in custody
- People who could be regarded as vulnerable
- People engaged in illegal activities (e.g., drug taking)
- Groups of people whose relationship among each other allow one to have influence over the other such as: Carers and patients with chronic conditions; teachers and their students; prison authorities and prisoners; employers and employees
- Groups where permission of a gatekeeper is normally required for initial access to members.

○ Yes ○ No ◉ N/A

Note: You may also need to obtain satisfactory clearance from the relevant authorities.

Does the project/assignment involve external funding or external collaboration where the funding body or external collaborative partner requires the University

○ Yes ◉ No ○ N/A



Note: You may also need to obtain satisfactory clearance from the relevant authorities.

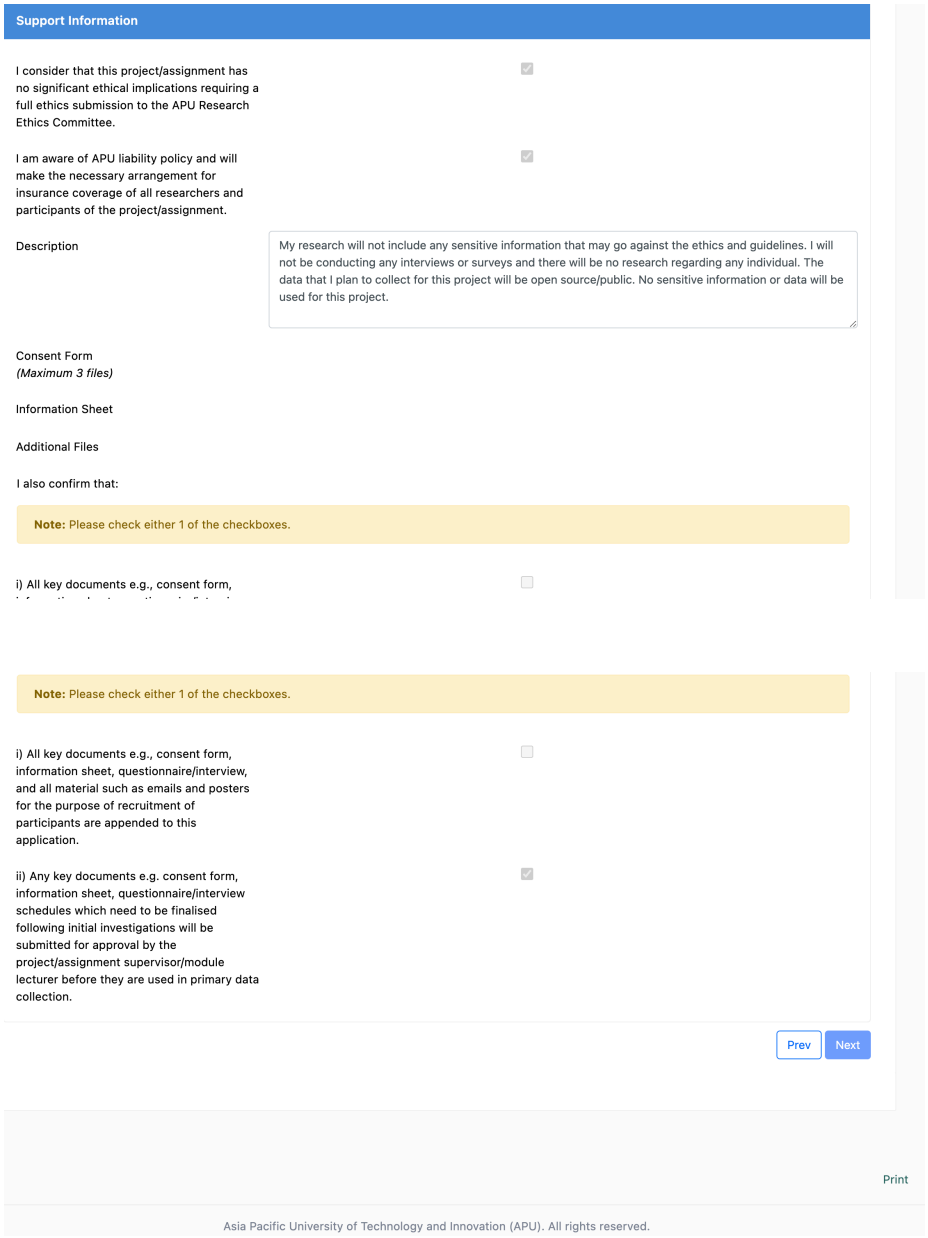
Does the project/assignment involve external funding or external collaboration where the funding body or external collaborative partner requires the University to provide evidence that the project/assignment had been subject to ethical scrutiny?

○ Yes ◉ No ○ N/A

Note: If you have ticked Yes to any of questions above, you should complete the Full Ethics Approval Form. There is an obligation on student and supervisor to bring to the attention of the APU School Research Ethics Committee any issues with ethical implications not clearly covered by the above checklist.
Full Ethics Form

Prev Next

Print


Support Information
I consider that this project/assignment has no significant ethical implications requiring a full ethics submission to the APU Research Ethics Committee.
I am aware of APU liability policy and will make the necessary arrangement for insurance coverage of all researchers and participants of the project/assignment.
Description
My research will not include any sensitive information that may go against the ethics and guidelines. I will not be conducting any interviews or surveys and there will be no research regarding any individual. The data that I plan to collect for this project will be open source/public. No sensitive information or data will be used for this project.
Consent Form
(Maximum 3 files)
Information Sheet
Additional Files
I also confirm that:
Note: Please check either 1 of the checkboxes.
i) All key documents e.g., consent form,
Note: Please check either 1 of the checkboxes.
i) All key documents e.g., consent form, information sheet, questionnaire/interview, and all material such as emails and posters for the purpose of recruitment of participants are appended to this application.
ii) Any key documents e.g. consent form, information sheet, questionnaire/interview schedules which need to be finalised following initial investigations will be submitted for approval by the project/assignment supervisor/module lecturer before they are used in primary data collection.
Prev
Next
Print
Asia Pacific University of Technology and Innovation (APU). All rights reserved.

3. Log Sheet

(APU: Serial Number)

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

1. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
2. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
3. A log sheet is to be brought by the STUDENT to each supervisory session.
4. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
5. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
6. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.
7. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: ABDUL MUHAIMIN AMAN Date: 22.04.2025 Meeting No: 1

Project title: Developing a Smart Traveller Insights System to Analyse Post Pandemic Tourism Recovery in Asia Pacific

Intake: APD3F2502CS(DA)

Supervisor's name:

Fatin Izzati binti Ramli **Supervisor's signature: Fatin**

Items for discussion (noted by student before mandatory supervisory meeting):

1. **Project Proposal**
2. **Algorithms to use**
3. **Software for pre-processing**

Record of discussion (noted by student during mandatory supervisory meeting):

1. **Addressed the algorithms to use for the project.**
2. **Addressed the pre-processing techniques**
3. **Supervisor agreed on the topic and the implementation.**

Action List (to be attempted or completed by student by the next mandatory supervisory meeting):

1. **Start gathering data for the project.**
2. **Finalize on the algorithms and pre-processing techniques**

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not for consultation, the project manager should be informed ONE (1) week prior t

(APU: Serial Number)

Project Log Sheet – Supervisory Session

Notes on use of the project log sheet:

8. This log sheet is designed for meetings of more than 15 minutes duration, of which there must be at minimum SIX (6) during the course of the project (SIX mandatory supervisory sessions).
9. The student should prepare for the supervisory sessions by deciding which question(s) he or she needs to ask the supervisor and what progress has been made (if any) since the last session, and noting these in the relevant sections of the form, effectively forming an agenda for the session.
10. A log sheet is to be brought by the STUDENT to each supervisory session.
11. The actions by the student (and, perhaps the supervisor), which should be carried out before the next session should be noted briefly in the relevant section of the form.
12. The student should leave a copy (after the session) of the Project Log Sheet with the supervisor and to the administrator at the academic counter. A copy is retained by the student to be filed in the project file.
13. It is recommended that students bring along log sheets of previous meetings together with the project file during each supervisory session.

14. The log sheet is an important deliverable for the project and an important record of a student's organisation and learning experience. The student **must** hand in the log sheets as an appendix of the final year documentation, with sheets dated and numbered consecutively.

Student's name: ABDUL MUHAIMIN AMAN Date: 07.05.2025 Meeting No: 2

Project title: Developing a Smart Traveller Insights System to Analyse Post Pandemic Tourism Recovery in Asia Pacific

Intake: APD3F2502CS(DA)

Supervisor's name:

Fatin Izzati binti Ramli Supervisor's signature: Fatin

Items for discussion (noted by student <u>before</u> mandatory supervisory meeting):

4. **Constraints of the project such as data gathering**
5. **Updating Supervisor on the report.**
6. **Report Formatting.**

Record of discussion (noted by student <u>during</u> mandatory supervisory meeting):

4. **Addressed how gathering data of multiple cities is difficult.**
5. **Addressed the order of sub-topics in the report.**

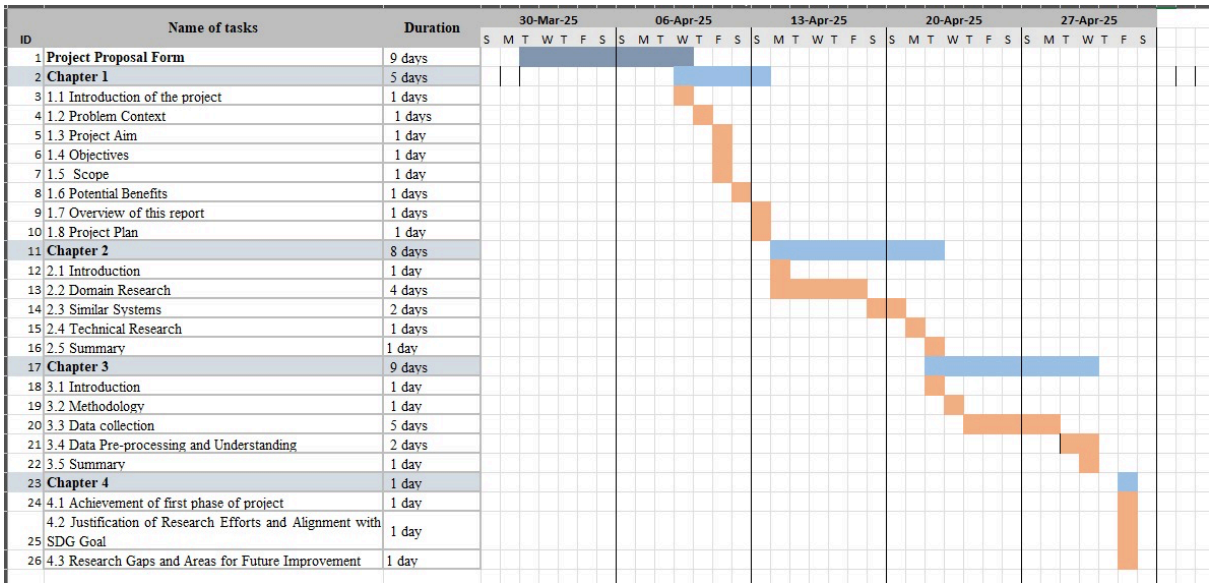
Action List (to be attempted or completed by student by the <u>next</u> mandatory supervisory meeting):

3. **Finish up on data visualisation.**
4. **Submit report 1.**

Note: A student should make an appointment to meet his or her supervisor (via the consultation system) at least ONE (1) week prior to a mandatory supervisor session – please see document on project timelines. In the event a supervisor could not be booked for consultation, the project manager should be informed ONE (1) week prior

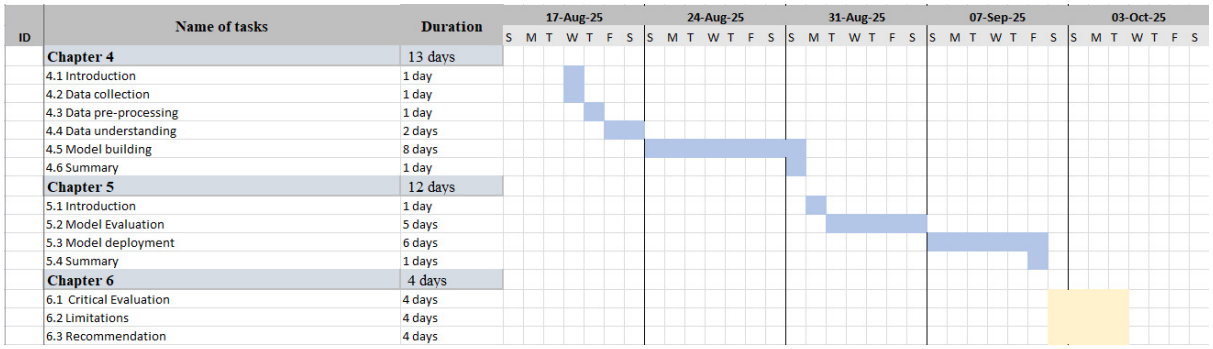
4. Gantt chart

Semester 1



ID	Name of tasks	Duration
1	**Project Proposal Form**	9 days
2	**Chapter 1**	5 days
3	1.1 Introduction of the project	1 days
4	1.2 Problem Context	1 days
5	1.3 Project Aim	1 day
6	1.4 Objectives	1 day
7	1.5 Scope	1 day
8	1.6 Potential Benefits	1 days
9	1.7 Overview of this report	1 days
10	1.8 Project Plan	1 day
11	**Chapter 2**	8 days
12	2.1 Introduction	1 day
13	2.2 Domain Research	4 days
14	2.3 Similar Systems	2 days
15	2.4 Technical Research	1 days
16	2.5 Summary	1 day
17	**Chapter 3**	9 days
18	3.1 Introduction	1 day
19	3.2 Methodology	1 day
20	3.3 Data collection	5 days
21	3.4 Data Pre-processing and Understanding	2 days
22	3.5 Summary	1 day
23	**Chapter 4**	1 day
24	4.1 Achievement of first phase of project	1 day
25	4.2 Justification of Research Efforts and Alignment with SDG Goal	1 day
26	4.3 Research Gaps and Areas for Future Improvement	1 day

Semester 2

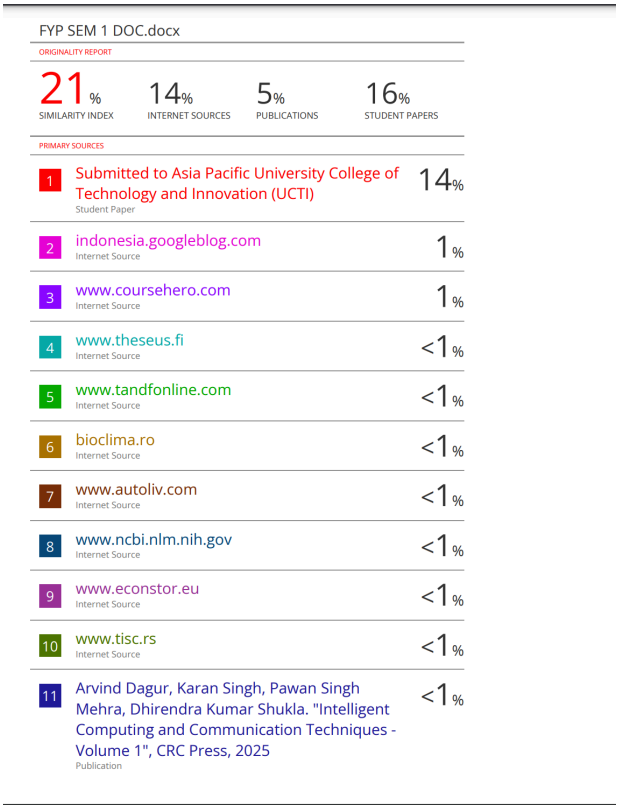


ID	Name of tasks	Duration
	Chapter 4	13 days
	4.1 Introduction	1 day
	4.2 Data collection	1 day
	4.3 Data pre-processing	1 day
	4.4 Data understanding	2 days
	4.5 Model building	8 days
	4.6 Summary	1 day
	Chapter 5	12 days
	5.1 Introduction	1 day
	5.2 Model Evaluation	5 days
	5.3 Model deployment	6 days
	5.4 Summary	1 days
	Chapter 6	4 days
	6.1 Critical Evaluation	4 days
	6.2 Limitations	4 days
	6.3 Recommendation	4 days

Figure 58 Gantt chart

5. Turnitin

   Semester 1



FYP SEM 1 DOC.docx

ORIGINALITY REPORT

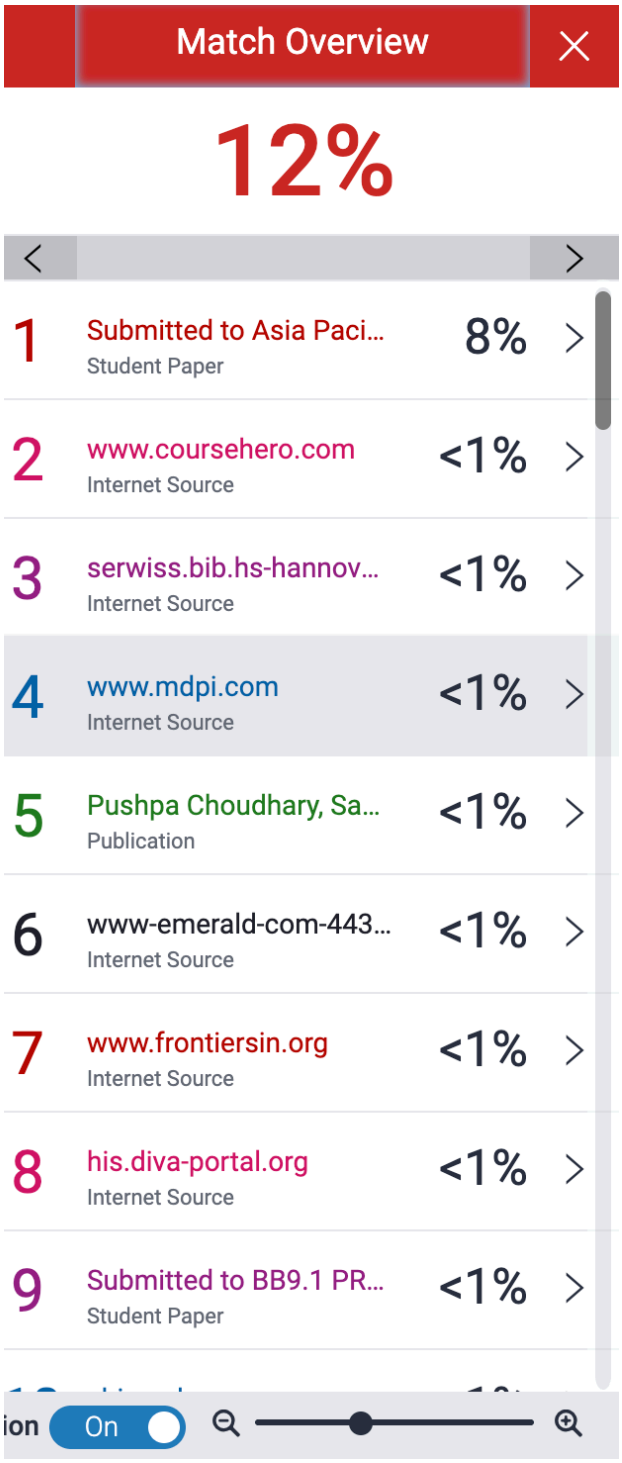
21%	14%	5%	16%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	14%
2	indonesia.googleblog.com Internet Source	1%
3	www.coursehero.com Internet Source	1%
4	www.theseus.fi Internet Source	<1%
5	www.tandfonline.com Internet Source	<1%
6	bioclima.ro Internet Source	<1%
7	www.autoliv.com Internet Source	<1%
8	www.ncbi.nlm.nih.gov Internet Source	<1%
9	www.econstor.eu Internet Source	<1%
10	www.tisc.rs Internet Source	<1%
11	Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dhirendra Kumar Shukla. "Intelligent Computing and Communication Techniques - Volume 1", CRC Press, 2025 Publication	<1%

12	Submitted to Nottingham Trent University Student Paper	<1 %
13	www.mendeley.com Internet Source	<1 %
14	Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dhirendra Kumar Shukla. "Intelligent Computing and Communication Techniques - Volume 2", CRC Press, 2025 Publication	<1 %
15	Wenjie Han, Yong Li, Yunpeng Li, Tao Huang. "A deep learning model based on multi-source data for daily tourist volume forecasting", Current Issues in Tourism, 2023 Publication	<1 %
16	api.mziq.com Internet Source	<1 %
17	online.op.ac.nz Internet Source	<1 %
18	Submitted to BPP College of Professional Studies Limited Student Paper	<1 %
19	dspace.bu.ac.th Internet Source	<1 %
20	www.slideshare.net Internet Source	<1 %
21	Submitted to University of Melbourne Student Paper	<1 %
22	Wei Su, Yuhao Han, Xu Jin, Zhongyan Liu, Xiaosong Zhang. "Energy, economic, and environmental evaluation of a GAX-based cross-type absorption-resorption heat pump", Energy Conversion and Management, 2024	<1 %

Semester 2


Match Overview
12%
1 Submitted to Asia Paci... 8%
Student Paper
2 www.coursehero.com <1%
Internet Source
3 serwiss.bib.hs-hannov... <1%
Internet Source
4 www.mdpi.com <1%
Internet Source
5 Pushpa Choudhary, Sa... <1%
Publication
6 www-emerald-com-443... <1%
Internet Source
7 www.frontiersin.org <1%
Internet Source
8 his.diva-portal.org <1%
Internet Source
9 Submitted to BB9.1 PR... <1%
Student Paper
ion On