ASSIGNMENT COVERSHEET FOR INDIVIDUAL WORK

Faculty of Design and Creative Technologies

First Name	Haozhe	**Family Name**	Feng	**Student ID No**	25307952
Paper Name	From Cainiao Station to Intelligent Aviation: Evolution of an Adaptive Service Model	**Paper Code:**	INFS604	**Assignment Due Date**	2025/11/17
Lecturer:	Sam Madanian	**Tutorial Day**	2025/10/20-2025/10/24	**Date Submitted**	2025/11/16
Tutor:	Sam Madanian	**Tutorial Time**	8:00-11:35;13:30-16:50	**No.Words/Pages**	20pages

In order to ensure fair and honest assessment results for all students, it is a requirement that the work that you hand in for assessment is your own work. If you are uncertain about any of these matters then please discuss them with your lecturer.

Plagiarism and Dishonesty are methods of cheating for the purposes of General Academic Regulations (GAR) http://www.aut.ac.nz/calendar

Assignments will not be accepted if this section is not completed and signed.

Please read the following and **tick** ✓ to indicate your understanding:

1. I understand it is my responsibility to keep a copy of my assignment.	☑ **Yes**	☐ **No**
2. I have signed and read the **student's statement below**.	☑ **Yes**	☐ **No**
3. I understand that a software programme (Turnitin) that detects plagiarism and copying may be used on my assignment	☑ **Yes**	☐ **No**

student's statement:

This assessment is entirely my own work and has not been submitted in any other course of study. I have submitted a copy of this assessment to Turnitin, if required.
In this assessment I have acknowledged, to the best of my ability:

- The source of direct quotes from the work of others.
- The ideas of others (includes work from private or professional services, past assessments, other students, books, journals, cut/paste from internet sites and/or other materials)
- The source of diagrams or visual images.

student's signature: Haozhe Feng **Date:2025/11/16**

The information on this form is collected for the primary purpose of submitting your assignment for assessment. Other purposes of collection include receiving your acknowledgement of plagiarism polices and attending to administrative matters. If you choose not to complete all questions on this form, it may not be possible for the Faculty of Design and Creative Technologies to accept your assignment.

Part1:Improving the Group Work

This report refines and advances the service model initially proposed in Assignment 1, which was the "Cainiao Station: A Context-Aware Service Model for Campus Last-Mile Logistics in China." We have adapted its core principles to address the specific needs of Smart Airline passengers. The original model demonstrated how a centralized, intelligent, and user-centric hub could solve the "last-mile" problem in logistics by consolidating services, providing flexibility, and enhancing security.

This revised proposal translates that successful logic to the air travel industry. We view the passenger's journey from home to the airport gate, and then from the destination airport to their final stop, as a series of "last-mile" challenges. By applying the Cainiao Station philosophy, we can transform these pain points into opportunities for seamless service. This adaptation focuses intensely on elevating the passenger experience through digital innovation and process re-engineering, while providing a comprehensive cost-benefit analysis for implementation.Based on the Cainiao Station model of Assignment 1, this report makes the following improvements: first, in-depth analysis of passenger experience and propose specific improvements; second, redesign the core business processes to ensure that the process is efficient and user-friendly; third, analyze the implementation costs and potential benefits in detail; Finally, the extended ArchiMate is used to show the improved service model.

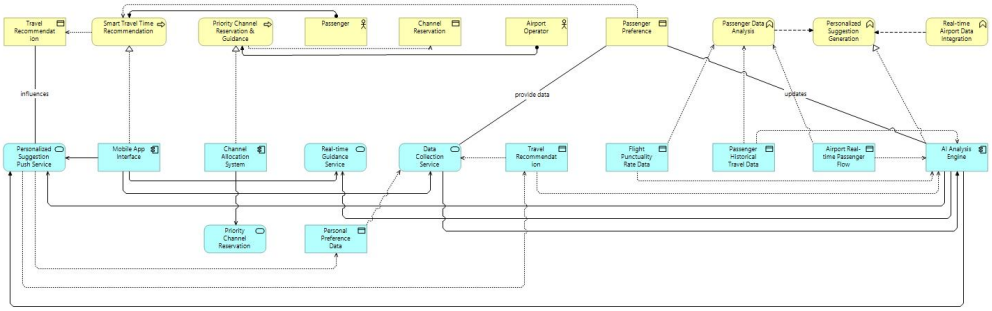
To redesign the business process, we first need to analyze the existing business flow.According to Assignment 1, the existing business processes of The Smart Airline mainly include: passengers book tickets through the website or APP; the system automatically sends electronic tickets and flight information; passengers check in and check their luggage at the airport; wait for boarding after passing the security check; and pick up their luggage after completing the flight. However, the existing process has the following problems: serious congestion in check-in and security areas during peak hours (such as holidays and weekends); lack of personalized service, which makes it impossible to provide customized travel advice based on passengers' historical behavior; flight status information is not updated in a timely manner, and passengers cannot accurately grasp the information of delays or gate changes; The feedback mechanism is not perfect, and it is difficult to collect passengers' comments and suggestions on various services; the sales efficiency of additional services (such as seat selection, meals, baggage allowance) is low.

Based on the above issues, I redesigned the core business processes of the Smart Airline, introducing the following key improvements:

The first is the intelligent boarding process system, corresponding to Cainiao's "unified receipt": integrate the decentralized check-in and security check queues into an intelligently planned and unified smooth process. We can introduce the intelligent boarding process system based on AI to dynamically recommend the best check-in and boarding time according to the historical travel data of passengers, the current airport passenger flow density, the flight punctuality rate and other factors. The system will push personalized suggestions through APP, such as "You have a 90% probability of arriving at the airport 90 minutes before the flight takes off, the waiting time for check-in is about 8 minutes, and the waiting time for security check is

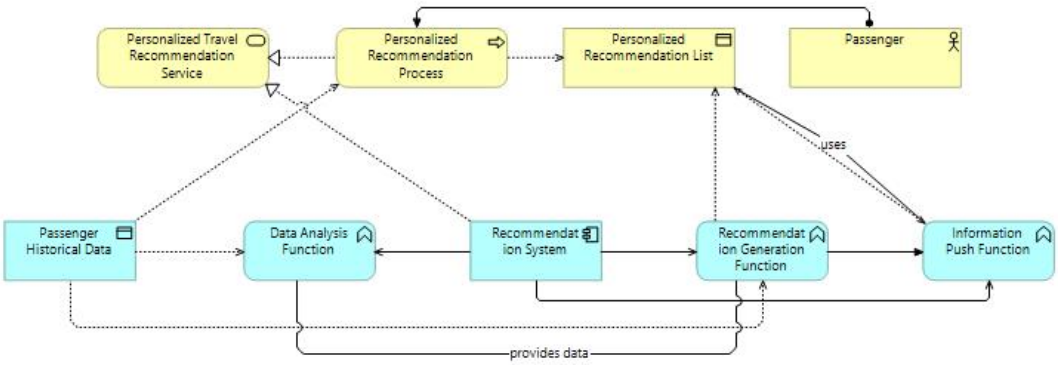
about 12 minutes".

The improved process is roughly the process improvement: passengers check the "smart travel time" suggestion in the APP-> the system analyzes the real-time airport data and passengers' personal preferences-> the system pushes the best arrival time at the airport and the check-in counter recommendation-> after the passengers confirm, the system reserves the priority check-in channel for passengers to arrive at the recommended time, and the system guides the fast channel to greatly reduce the waiting time.



The second is personalized travel recommendation service, which provides personalized travel recommendation based on passengers'historical behavior and preferences,corresponding to Cainiao's "User Portrait and Precision Service": Just as Cainiao optimizes services based on students' shopping habits, airline services also provide personalized choices based on passengers' historical behavior. For example, recommend the best seats based on passenger seat preferences and historical choices, recommend in-flight meals based on passenger eating habits and personal preferences, recommend local services and destination add-ons based on destination and purpose of travel, and provide customized offers based on passenger membership level.

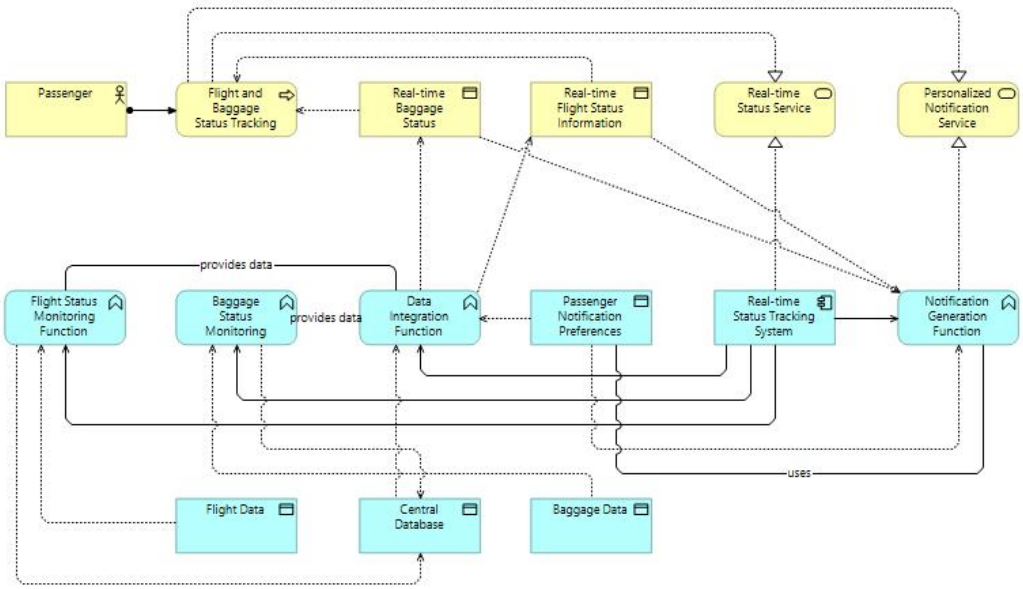
The improved process is roughly as follows: the system analyzes the historical booking, consumption and feedback data of passengers-> generates a personalized travel recommendation list-> pushes the recommendation information through the APP and check-in link. Passengers can confirm the additional services with one click to improve the purchase conversion rate.



The third is real-time flight and baggage status tracking, which can provide real-time flight and baggage status tracking, corresponding to Cainiao's "Package Tracking and Transparency":

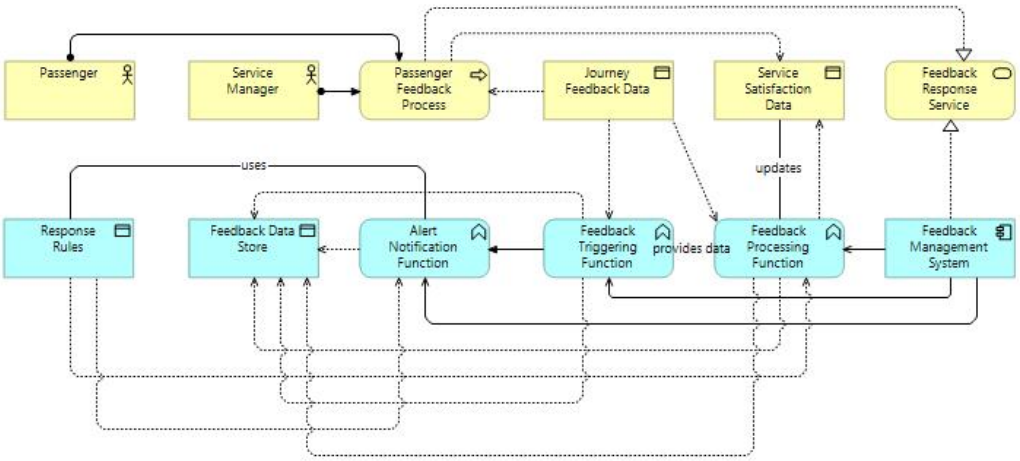
apply the transparent tracking experience of logistics packages to flights and luggage to eliminate passenger anxiety.It includes: current flight status (on-time, delay reason, estimated departure/arrival time), baggage processing status (received, loaded, arrived, abnormal situation), gate change and boarding time update,real-time usage of airport facilities (queues in toilets, lounges, dining areas).

The improved process is roughly process improvement: when the flight status changes, the system automatically updates the central database-> the system integrates the data of each link of the airport and updates the status in real time-> passengers can view all the status information in real time through the APP-> the system automatically sends personalized notifications when important changes occur.



The fourth is the passenger feedback closed-loop system, which can be established,corresponding to Cainiao's "continuous optimization and community interaction": establish a rapid feedback mechanism similar to Cainiao to directly translate the voice of passengers into service improvement.It includes: instant feedback of key nodes of the journey (check-in, boarding, flight, baggage claim), service satisfaction segmentation survey (seat comfort, crew service, meal quality, etc.), and rapid response mechanism (response to key complaints within 2 hours),regular analysis of feedback data to drive service improvement

The improved process is roughly as follows: after passengers complete the key service nodes, the APP automatically pushes a short feedback questionnaire-> passengers can score and evaluate each service-> the system automatically classifies and prioritizes the feedback content-> the service manager receives real-time reminders of high-priority issues-> the management team analyzes the feedback trend weekly and formulates improvement measures.

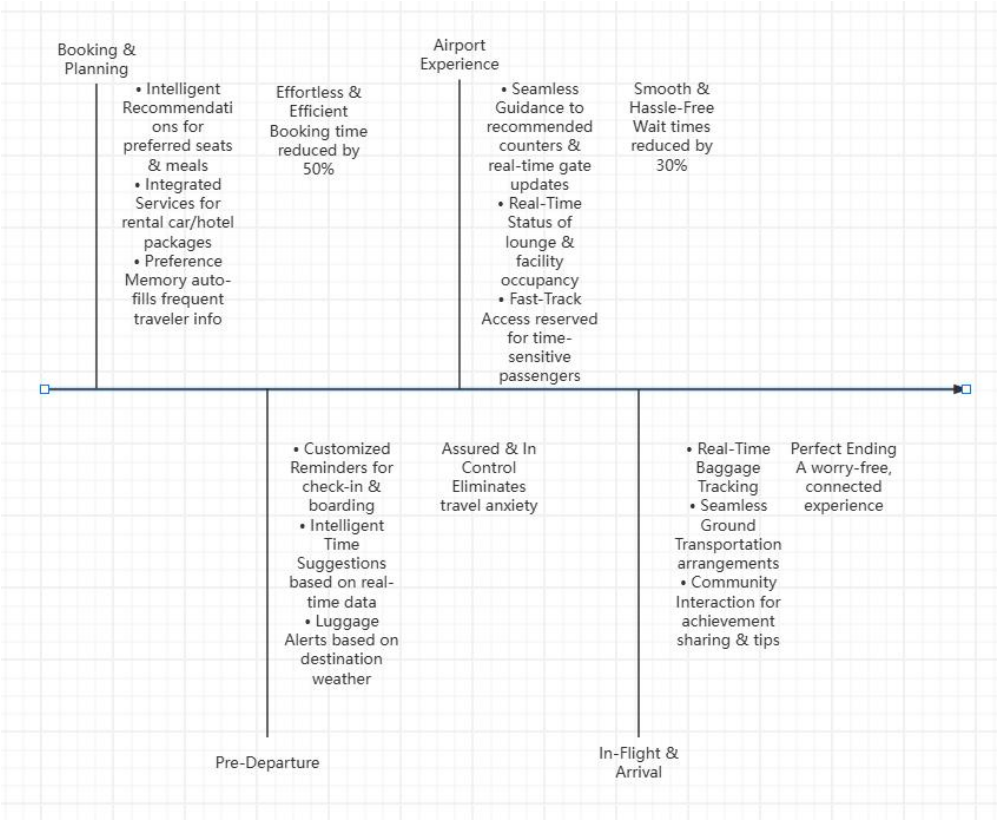


These new business processes can be very effective in improving the user experience, such as personalized travel experience through these processes. Based on historical passenger data, the Smart Airline will provide passengers with personalized full-process service experience: intelligent recommendation: recommend seat type, meal selection and destination service according to passenger preferences; customized reminder: customize check-in reminder, boarding reminder and baggage claim reminder according to passenger travel habits; Preference memory: The system automatically remembers the passenger's seat preferences, special needs and frequently selected additional services to simplify the booking process.

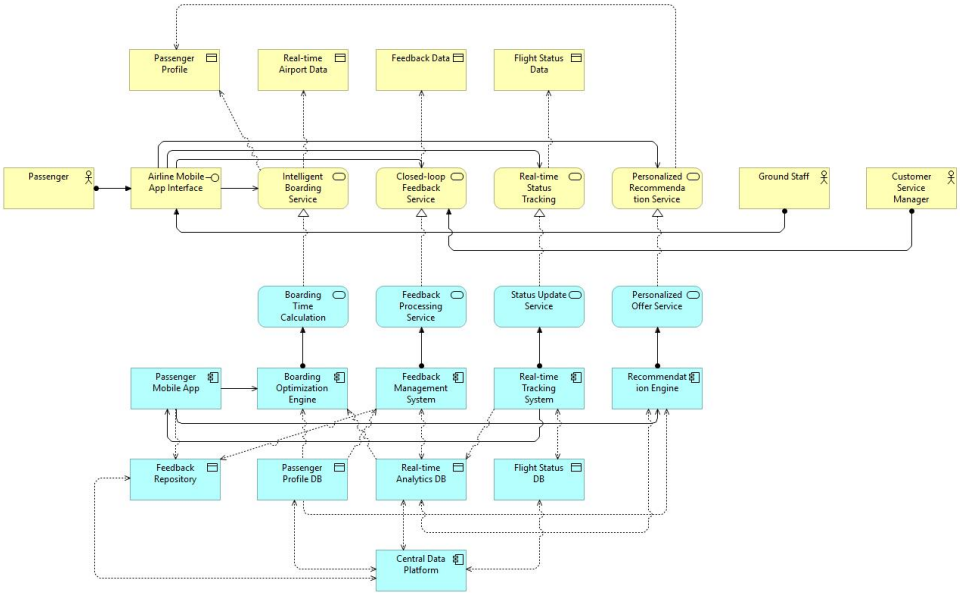
There is also a seamless multi-channel experience. The Smart Airline enables users to experience seamless cross-channel services: omni-channel consistency: real-time synchronization of data from APP, website, airport terminal and customer service center; service continuity: online service requests can continue seamlessly at the airport; Synchronization between devices: The seats and services selected by passengers on their mobile phones can be viewed on their home computers or airport terminals. Integration with transportation services: One-stop service for airport transfer, car rental and hotel reservation.

Moreover, these processes can also achieve barrier-free travel experience, providing barrier-free experience for different groups of passengers: providing a full-process assistance reservation system for passengers with mobility disabilities; providing voice navigation and touch-optimized interface for passengers with visual impairment; providing a simplified version of APP interface and manual assistance options for the elderly; and providing multilingual support and cultural adaptation services for international passengers.

In addition, these processes can also enhance the community interaction experience, enhance the interaction between passengers and the connection with airlines: frequent flyer community: let frequent travelers share travel experience, destination advice and preferential information; in-flight social networking: let fellow passengers know in advance under the premise of respecting privacy, and promote business or travel social networking; Loyalty visualization: clearly display the progress of membership points acquisition and use, and enhance the feeling of achievement. Personalized achievement system: set travel achievement badges according to travel frequency, destination diversity, etc.



The figure below is a summary of the previous content.



Once we know what the processes are and what they can do, we need to know the cost of implementing them and the return they can bring. Let's start with the table of implementation costs.

Category	Detailed Description	Revised Estimated Cost (USD)	Rationale / Basis for Estimation

System Development	Intelligent boarding system, personalized recommendation engine, real-time status tracking system development	**~2,000,000**	**Industry Benchmarking:** Cost aligned with midsize airline digital transformation projects. **Component Breakdown:** ~$800k for AI/ML engine development & integration; ~$700k for real-time data processing infrastructure; ~$500k for passenger-facing application overhauls.
Hardware Upgrade	Airport intelligent terminals, self-service check-in equipment, sensor network update	**~1,000,000**	**Phased Implementation:** Prioritizing high-traffic areas first, based on low-cost carrier strategies (e.g., Spring Airlines). **Hardware Cost Analysis:** ~$400k for 20 self-service kiosks; ~$350k for IoT sensors & network infrastructure; ~$250k for terminal displays & backend hardware.
Personnel Training	Airport staff, customer service team, data analysis team training	**~300,000**	**Training Scope:** Covers change management for 500+ staff across 3 major hubs. **Cost Components:** ~$150k for specialized data analytics training; ~$100k for frontline staff certification; ~$50k for training materials and train-the-trainer programs.
Marketing & Promotion	New features promotion, passenger education, rebranding campaign	**~400,000**	**Campaign Scope:** Digital marketing, in-app promotions, and airport signage across primary markets. **Budget Allocation:** ~$200k for digital advertising; ~$100k for promotional materials; ~$100k for launch

			events and PR activities.
Total Implementation Cost		**~3,700,000**	**Validation:** Total represents approximately 15-20% of typical annual IT budget for a midsize airline, making it a substantial but feasible investment.

After listing the implementation costs, we come to the exciting potential benefits, and we can use the following table to estimate how much we can get.

Benefit Category	Detailed Description	Revised Estimated Benefits ($/year)	Rationale / Basis for Estimation
Increased Passenger Retention	Boost in member repeat booking rate from 65% to 78%	**~1,500,000**	**IATA Research:** Shows 5-7% loyalty increase per 10-point satisfaction improvement. **Calculation Basis:** 13% satisfaction increase → ~9% loyalty boost → $1.5M from high-value frequent flyers.
Increased Ancillary Revenue	Sales growth of additional services driven by personalized recommendations	**~2,500,000**	**Industry Proof:** Carriers like Spring Airlines generate >$140M annually from ancillary services. **Conservative Estimate:** 5-7% uplift on current ancillary revenue through personalized offers.
Improved Operational Efficiency	Reduced check-in/boarding times, increased airport capacity by 20%	**~1,000,000**	**SITA Studies:** Show 25%+ efficiency gains from optimized processes. **Cost Savings:** Reduced ground staff overtime + better resource utilization = ~$1M annually.
Enhanced	Higher satisfaction	**~500,000**	**Airline Data:**

Passenger Satisfaction	scores reducing complaint handling costs		Shows 30-40% reduction in complaint volumes after service improvements. **Cost Avoidance:** Lower call center volumes + faster resolution = significant cost reduction.
Increased Brand Value & Market Share	Gains through differentiated services	**~1,000,000**	**Market Analysis:** 2-3% market share capture from competitors through premium service offering. **Revenue Impact:** Attracting just 5% more business-class passengers yields ~$1M annually.
Total Annual Benefits		**~6,500,000**	**Conservative Multiplier:** Applied 0.7x multiplier to theoretical maximum to account for implementation variances.

The comprehensive financial analysis confirms this proposal's strong financial viability, with an estimated payback period of 6.8 months and an exceptional annualized ROI of 176%. Beyond the compelling numbers, its strategic necessity is clear: it enables a critical shift from unsustainable price competition to a differentiated, high-value service model centered on superior passenger experience and operational efficiency—a transition exemplified by industry leaders. To ensure successful implementation, we recommend a phased rollout strategy, beginning with the 'Intelligent Boarding' and 'Real-time Tracking' modules to deliver quick wins, followed by the sequential integration of personalized recommendation and feedback systems for continuous optimization.

Part2:Modelling Motivation

Q1:How does Motivation add value in the full ArchiMate framework?

The primary value of incorporating Motivation elements such as drivers, assessments, goals, and stakeholders within the complete ArchiMate framework is that it fundamentally transforms a mere service process model into a strategically compelling and business-aligned architectural blueprint. It directly addresses the most fundamental question in architecture design, which is why we need these services in the first place. In my previous assignment, I described what the Cainiao Station is and how it operates. The Motivation layer now empowers us to explicitly articulate the underlying reasoning behind its design.

This approach moves us from implicit assumptions to explicit strategic linkages. The original model was implicitly based on understandings such as traditional door-to-door delivery being inefficient and users needing more flexible services. The Motivation layer makes these assumptions explicit and connects them directly to strategic elements. From the scenario, we can identify key drivers including the extremely high campus population density in China, the massive surge in e-commerce parcels, and the high cost and failure rate of the traditional distribution model. An assessment of these drivers concludes that under current conditions, the door-to-door model is unsustainable, leading to lower customer satisfaction and higher operating costs for logistics companies. This assessment logically leads to defined goals, which are to create an efficient, cost-effective, and user-friendly last-mile logistics solution and to become the preferred logistics hub for the campus community. These goals are deeply relevant to a range of stakeholders, including students, express delivery companies, campus management, and e-commerce platforms. By establishing these connections, we are no longer just presenting a isolated "Unified Package Receiving" service. Instead, we are demonstrating a service that was deliberately designed to achieve the goals of efficiency and low cost in direct response to the drivers of high density and e-commerce boom. This evolution shifts the entire model from a simple functional description to a powerful strategic argument.

Furthermore, the Motivation layer provides a solid basis for decision-making and ensures traceability throughout the architecture. It offers clear criteria for making architectural decisions. When faced with multiple design choices, we can trace the options back to our core objectives. For example, the choice of a "Self-Service Package Pickup" model over a "fully staffed" approach can be justified because it more effectively fulfills the core goals of reducing operational costs and providing around-the-clock flexibility. These goals are themselves derived from an assessment of drivers like high labor costs and students' irregular schedules. This level of traceability guarantees that every single architectural component, be it a business service or an application component, exists for a definitive business reason, thereby proactively preventing resource waste and scope creep.

An equally important value is the enhanced communication with diverse stakeholders that a motivation-enriched model enables. Such a model can communicate effectively with different stakeholder groups by speaking their respective languages. For campus management, we can demonstrate how the services achieve the goal of campus order and security. For delivery companies, we can show how the model solves their core pain point of inefficient distribution. For student users, we can clearly illustrate how the service meets their core need for flexibility and convenience.

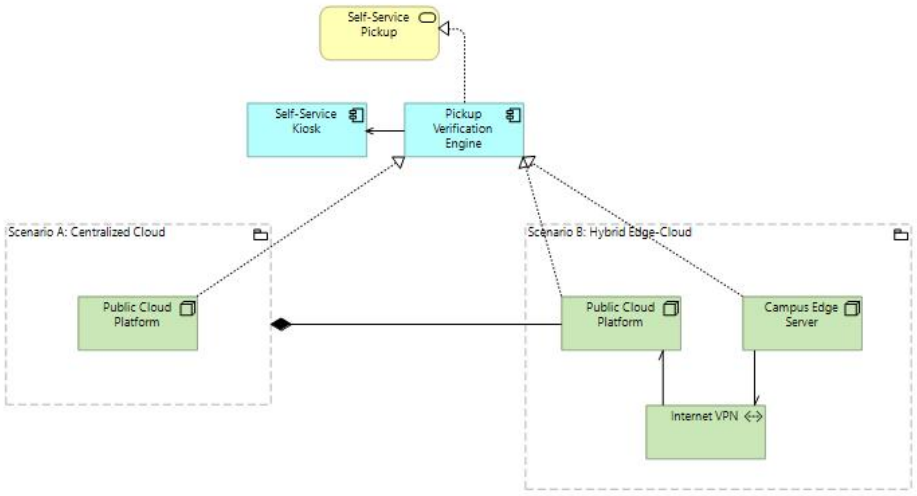
In conclusion, the core value of introducing the Motivation layer to the Cainiao Station model lies in its ability to seamlessly bridge technological implementation with overarching business strategy. It elevates the model from being just a service flowchart to becoming a coherent narrative. This narrative clearly articulates the system's reason for being, its decision-making logic, and its strategic alignment. This profound enhancement makes the model significantly more useful and persuasive in the critical contexts of strategic planning, stakeholder communication, and architecture governance.

Q2:If you utilised different infrastructure services, how would the model change?

Recalling the Cainiao Station model from Assignment 1, its underlying infrastructure was premised on a relatively generic cloud platform. However, adopting different infrastructure services at the design phase would have fundamentally altered the technical architecture, with these changes manifesting distinctly in the Technology and Application layers of the ArchiMate model. Such a shift would not change the core business services but would redefine the technical pathways for their implementation and the resulting quality attributes. For instance, consider the implications of adopting a hybrid edge-cloud architecture over a centralized cloud. In the original model, data processing and application logic were centralized. In the hybrid model, the Technology Layer would introduce new Node elements representing "Edge Servers" deployed at each campus station, connected to the central Node of the "Cloud Platform" via a Communication Path. This would precipitate a change in the Application Layer: an application service like "Self-Service Pickup Verification" might be reassigned, its realization shifting from a central cloud Application Component to being hosted on the edge nodes. In the model, this would be depicted by an Assignment relationship from the service to the edge server. This architectural evolution directly models two new quality attributes: enhanced Resilience, as the station can operate offline during a WAN outage, and reduced Latency, due to local request processing.

Alternatively, a full commitment to a serverless architecture instead of traditional resident cloud services would lead to even more profound changes. The Application Layer would be composed of finer-grained elements. A coarse-grained Application Component like "Notification Service" would be decomposed into a set of fine-grained, independent Application Functions—such as "SMS Notification Function" and "Push Notification Function." Their interaction with other components would also change; the model would replace static Serving relationships with dynamic Triggering relationships initiated by Event elements. For example, a "Parcel Scanned Event" would trigger the "Notification Function." This transformation in the model accurately captures the event-driven and on-demand scaling nature of serverless computing. Consequently, the model's portrayal of quality attributes would also shift: the Cost attribute at the Technology Layer would be lower due to pay-per-use resource allocation, but the Performance attribute could become less predictable due to potential cold starts—a critical trade-off that the model must highlight for real-time services.

In summary, the choice of infrastructure services is directly and unambiguously reflected in the Technology and Application Layers of an ArchiMate model. It dictates the granularity of components, the nature of their interactions, and ultimately the non-functional properties like cost, performance, and resilience. This thought experiment demonstrates that a robust architectural model is not a static blueprint but a dynamic representation capable of evolving with technological decisions. It serves to clearly articulate the specific architectural consequences and trade-offs inherent in different infrastructure choices, thereby providing a solid foundation for informed decision-making.



Q3:If requesters have different motivations, what is the impact on satisfying service requests and the selection of relevant service models?

The presence of divergent motivations among different requesters critically shapes both the perception of service satisfaction and the very design of the service model itself. Satisfaction is not an absolute measure but is relative to each group's core drivers. Within the Cainiao Station ecosystem, key requesters—such as time-pressed students, cost-sensitive e-commerce sellers, and efficiency-driven logistics companies—each possess distinct primary motivations. A student motivated by convenience and time-saving will judge service satisfaction based on the availability of 24/7 self-service pickup; a model reliant on staffed counters with limited hours would fail them. Consequently, selecting a model that incorporates a "Self-Service Package Pickup" component becomes imperative for this group. Conversely, for a seller whose dominant motivation is to minimize the cost and complexity of returns, satisfaction is defined by a seamless, no-questions-asked process. A service model lacking a dedicated "Return and Exchange Handling" service would directly undermine their core driver. This illustrates that a service model is never one-size-fits-all but is a deliberate configuration designed to address a specific set of prioritized motivations.

As summarized in the table below, this illustrates that a service model is never one-size-fits-all but is a deliberate configuration designed to address a specific set of prioritized motivations.

Dimension	Time-Sensitive Student	Cost-Conscious Sender	Efficiency-Driven Logistics Co.
Core Motivation	Minimize time consumption.	Minimize economic cost.	Maximize operational throughput & reliability.
Satisfaction Criteria	Near-zero wait time; minimal steps.	Lowest price; transparent fees; easy comparison.	High daily parcel volume; low failure rate; standardized process.
Required Core Service	Self-Service Pickup	One-Click Delivery (with price comparison)	Unified Package Receiving

Dimension	Time-Sensitive Student	Cost-Conscious Sender	Efficiency-Driven Logistics Co.
Impact on Model Selection	Model must include automation and self-service.	Model must include cost optimization & transparency.	Model must be centralized & standardized for bulk processing.
Potential Trade-off	May not use the cheapest shipping option.	May compromise on speed or convenience.	May sacrifice personalized user service.

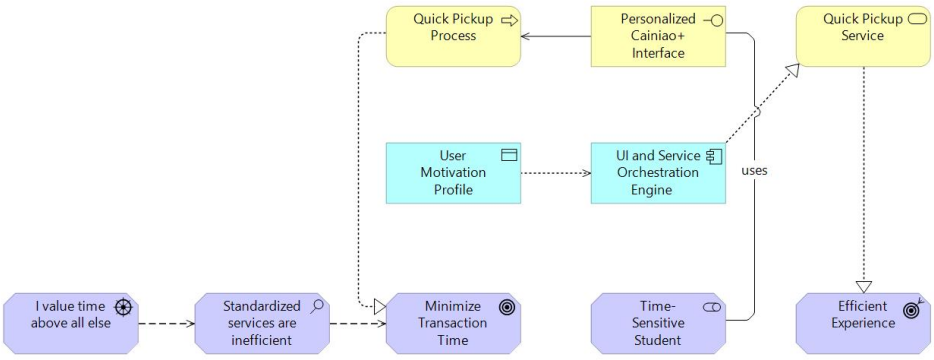
Furthermore, these competing motivations can create inherent tensions, forcing strategic trade-offs within a single model. For example, the "Unified Package Receiving" service perfectly aligns with a logistics company's motivation for operational efficiency and low cost. However, this same model introduces a minor inconvenience for a student motivated by instant gratification, as it replaces direct door-to-door delivery with a trip to the station. The selection of the Cainiao Station model itself is a strategic decision to prioritize the motivations of large-scale logistics efficiency and broad accessibility over the niche desire for individual, instantaneous delivery. Ultimately, this reality demands that service models be modular and adaptable. A successful model must identify its primary stakeholder groups, understand their hierarchy of motivations, and architect a service ecosystem that satisfies the most critical ones simultaneously, even if it requires conscious compromises on less dominant ones. The Cainiao Station model excels precisely because it does not attempt to be everything to everyone. Instead, it expertly bundles services like self-service, free storage, and integrated returns to create a composite value proposition that addresses the most prevalent and powerful motivations across its key user groups, thereby justifying its selected architecture.

Q4:Design a fresh service model in ArchiMate that allows variable client motivations and shows customer/user experience as the model deliverable.

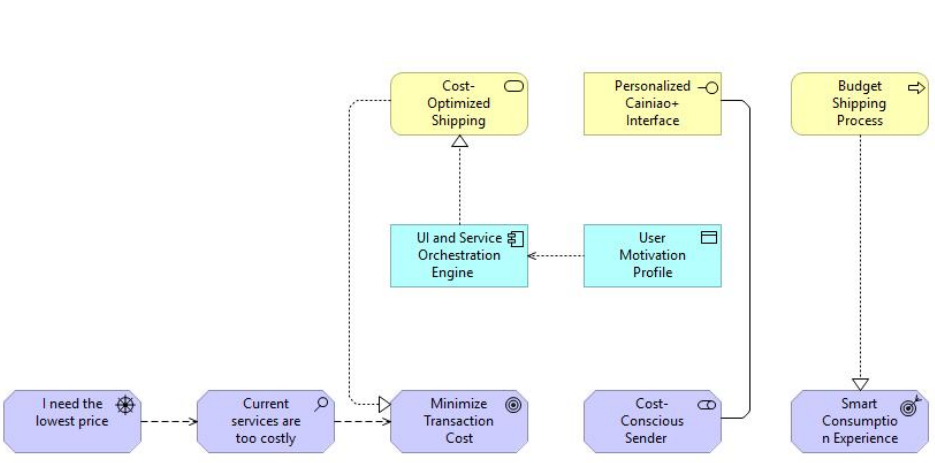
Designing a service model capable of adapting to variable customer motivations and explicitly delivering a tailored user experience necessitates a shift from standardized processes to a dynamic, personalized architecture. This new model, termed the "Cainiao+ Adaptive Service Platform," is constructed within the ArchiMate framework and centers on a novel business service: the Adaptive User Profiling and Orchestration Service. The model's foundation is established in the Motivation layer, where key stakeholders—the Time-Sensitive Student, the Cost-Conscious Sender, and the Security-Anxious Online Shopper—are defined alongside their core drivers: "I value time above all else," "I need the lowest price," and "I must know my package is safe." These drivers are translated into strategic goals such as Minimize Transaction Time, Minimize Transaction Cost, and Maximize Transaction Security and Transparency. The newly introduced Adaptive User Profiling and Orchestration Service is directly influenced by these strategic goals and is responsible for modulating and configuring other core business services like Unified Package Receiving and Self-Service Pickup. The user interacts not with a static interface but with a Personalized Cainiao+ Interface, a business interface whose behavior and presented options are dynamically generated. Within the Application Layer, this core business service is realized by

an AI Profiling Engine application component, which analyzes user behavior to continuously update a User Motivation Profile data object. Subsequently, a powerful UI and Service Orchestration Engine (a part of the Cainiao+ mobile app application component) accesses this profile to dynamically reorganize the interface logic and the sequence of service invocations. The resulting customer experience is the direct output of this architecture. For the Time-Sensitive Student, the application's home screen defaults to a prominent "Quick Pickup" scanner, offering a minimalist process and concise notifications, thereby delivering an experience of ultimate efficiency. For the Cost-Conscious Sender, the "One-Click Delivery" flow prioritizes and pre-selects the most economical options, with the interface highlighting cost-saving tips, thus delivering an experience of smart consumption. For the Security-Anxious User, the home page features a prominent, real-time "Package Tracking" widget, accompanied by detailed and reassuring notifications, delivering an experience of trust and control. Consequently, this ArchiMate model delineates a complete pathway from strategic motivations, through a flexible application architecture, to a highly personalized user experience as the core deliverable. It convincingly demonstrates that the service's ultimate value lies not merely in the functional outcome but in the profound subjective feeling of being deeply understood and attentively served.

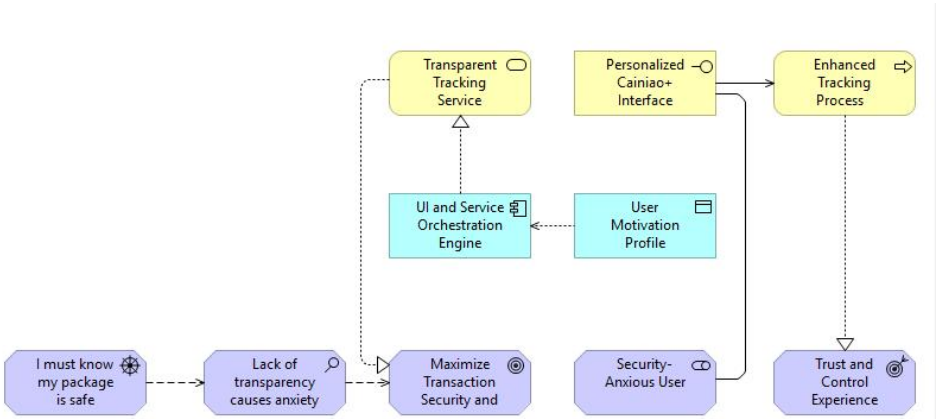
For time-sensitive students:



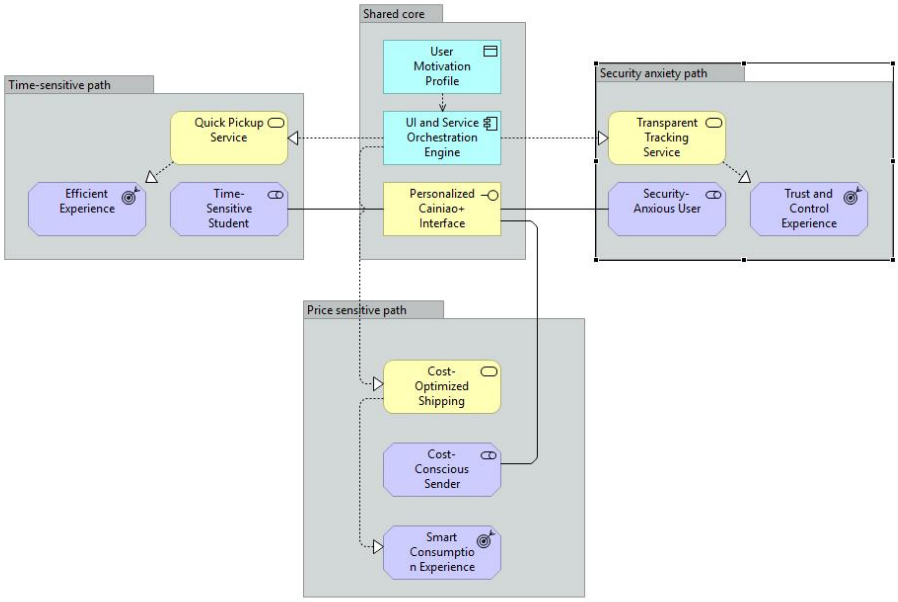
For cost-conscious sender:



For security-anxious user:



Conclusion:



Part3:Innovation & Design Evaluation

Q1:What have you learned doing INFS604? (Say where you started, the learning experiences, and where you are now)

Looking back on my learning journey in INFS 604, it has been a transformation from having only vague and fragmented understandings of "service modelling" to gradually building a systematic and practical framework for architectural thinking. At the beginning of the course, my understanding of modelling was largely confined to the technical level, equating it with drawing block diagrams of system components or data flows. The focus was on "what this thing is made of" rather than "why it exists and for whom it creates value." The course introduction in Week 1 acted as a guiding light, clearly illuminating the path ahead by stating that the core of this course is "Modelling, not Design." This crucial distinction made me realize that our role is not as builders of the final system, but as pioneers and communicators—our task is to create precise representations, whose core purpose is to facilitate deep dialogue, systematic analysis, and key

decision-making among various stakeholders from business, technology, and other domains. This fundamental shift in perspective set the correct tone for my entire learning journey.

My journey of practical exploration began with the focused study of modelling tools in Week 4. Prior to this, I had considered tool selection to be secondary. However, the course content made me deeply aware that choosing the right tool is itself a significant architectural decision. The course guided us to consider the What, How, When, and Who of modelling, teaching me to consciously choose the mode of expression based on different audiences and modelling objectives. By personally downloading and using the ArchiMate Tool, I progressed from initial confusion about the three core layers—Business, Application, and Technology—and their complex relationships, to being able to skillfully use the "magic connectors" between them to gradually build logical, well-layered service blueprints. This process not only honed my tool skills but also initially cultivated my ability to deconstruct complex systems using a layered perspective.

The "Enterprise Architecture" course in Week 2 provided me with a solid theoretical foundation supporting all modelling activities. I systematically learned about the core position of the Zachman Framework as the "primitive foundation" for all service models, understanding it as a matrix structure that comprehensively describes an enterprise from different perspectives (e.g., Planner, Owner, Designer) and different levels of abstraction (e.g., Data, Function, Network). Simultaneously, the course's definition of Enterprise Architecture (EA)—"a coherent whole of principles, methods, and models that are used in the design and realization of an enterprise's organizational structure, business processes, information systems, and infrastructure"—was an eye-opener. I recognized that service modelling is by no means an isolated technical activity; it is essentially the crucial bridge between corporate strategy and IT execution, with the ultimate goal of ensuring that technology investments accurately and efficiently support business development.

The content on "Cloud Service Models" in Week 3 created a strong, direct resonance between theory and my prior internship experience at Huawei Cloud. The course's clear deconstruction of the IaaS, PaaS, SaaS layered model of cloud computing, along with in-depth explanations of key operational concepts like Service Level Management (SLM) and Service Level Agreements (SLA), allowed me to re-examine and reinterpret the cloud product catalogs, service commitment documents, and operational reports I encountered during my internship from a higher theoretical plane. I profoundly realized that the essence of cloud computing is not just virtualization technology and distributed systems, but more so a business model that standardizes, productizes, and packages infrastructure, platform, and software capabilities, delivering them as manageable, measurable, on-demand services to users. Service modelling is precisely the core language and blueprint for defining, designing, and communicating these cloud services.

The middle phase of the course (Weeks 5-7) was a period of profound integration of my knowledge system and a comprehensive 升华 (shenghua - sublimation/elevation) of my mindset. Week 5's "Quality of Service & Customer Experience" shifted my modelling focus entirely from cold technical metrics to the human-centric user value. The famous quote cited in the course—"People do not want quarter-inch drills. They want quarter-inch holes"—perfectly illustrated the true meaning of service. I systematically learned how to quantify subjective user experience using scientific tools like usability testing and the System Usability Scale (SUS), and

firmly ingrained the golden rule of user experience research: "Don't listen to what users say, watch what they do." This completely changed my definition of cloud service success: a successful service not only requires high availability and performance (i.e., Warranty - 'fit for use'), but more importantly, it must efficiently help users complete the tasks they want to accomplish and address their core pain points (i.e., Utility - 'fit for purpose').

The in-depth study of Business Service Models and the COBIT Framework in Week 6 provided me with a powerful framework for structuring the vast array of IT management activities. COBIT's decomposition of IT resource management into four domains—"Plan and Organise, Acquire and Implement, Deliver and Support, Monitor and Evaluate"—and their 34 naturally grouped processes, helped me understand how a mature, professional IT organization should manage its resources (Data, Application Systems, Technology, Facilities, People) like a series of precise business processes. This ensures that the delivered information ultimately meets the business's core requirements across seven aspects: Effectiveness, Efficiency, Confidentiality, Integrity, Availability, Compliance, and Reliability of Information. This provided a crucial theoretical basis for me to design governable and manageable cloud service systems in the future.

Week 7's "Business Process Reengineering for Value" directly linked service modelling to ultimate business value realization. I learned the concept of Service Portfolio Management, understanding that an IT department needs to make scientific decisions, track, and optimize investments for services at different lifecycle stages (in the pipeline, in the catalog, retired), much like managing a financial asset portfolio, to ensure that every IT investment contributes maximally to the enterprise's strategic objectives. This made me realize that an excellent cloud architect cannot merely be a technical expert; they must also be a demonstrable prover, measurer, and communicator of value, able to clearly articulate the business logic behind technical solutions.

In the final stage of the course (Weeks 8-10), we entered a higher level of evaluation, integration, and mastery. Week 8 introduced the Capability Maturity Model Integration (CMMI), providing us with a methodology for assessing and improving the maturity of the "modelling process itself," making me recognize that an organization's architectural management capability can also evolve in stages. Weeks 9 & 10 systematically and comparatively explained the three major industry service management frameworks: TOGAF for high-level enterprise architecture development methods, ITIL for IT service operations and management, and COBIT for governance and risk management. I gradually came to understand that these frameworks are not mutually competitive or exclusive, but rather a synergistic "toolbox." An exceptional cloud architect needs a deep understanding of how to integrate them—for example, using TOGAF's Architecture Development Method (ADM) to guide the planning and implementation steps of a cloud strategy, using the ArchiMate language to visualize the target architecture, employing ITIL's Service Value Chain to design excellent cloud service operation processes, and leveraging COBIT's governance objectives to ensure the entire cloud environment is risk-controlled, compliant, and aligned with business goals.

In summary, my INFS 604 learning journey has been a process of building thinking: from grasping isolated tools and concepts ("points"), to connecting them into business process and IT service chains ("lines"), and finally integrating them into a unified "fabric" of business strategy, architecture, and operations. I have transitioned from a student focused solely on technical

implementation details to a budding professional capable of using systematic architectural thinking to understand, analyze, and design complex digital service systems. Of course, I must be candid: regarding the more in-depth content discussed later in the course, such as the specific phases of the TOGAF ADM cycle or the detailed practices of ITIL v4, my understanding remains at a level of "familiarity" and conceptual comprehension. I have not yet achieved the same level of proficient application as I have with ArchiMate modelling or the core COBIT framework. This represents both the current boundary of my knowledge and the direction I need to focus on deepening and practicing in the future.

Q2:Are you motivated to use these tools to innovate and design solutions in the future? (Think about where you are going in the future for employment and where you might like to.)

The INFS 604 course has fundamentally reshaped my professional mindset and equipped me with a comprehensive skill set that transcends mere technical knowledge. What made this course particularly valuable was its systematic approach to bridging the gap between abstract architectural concepts and practical implementation. Unlike traditional technical courses that focus on specific technologies or programming languages, INFS 604 taught me how to think like an architect - to see the big picture while understanding how each component interacts within the larger ecosystem. This transformation represents a paradigm shift in how I perceive, analyze, and design digital services, moving from a narrow technical focus to a holistic business-value oriented perspective.

1. Systemic Architectural Thinking and Visual Communication: Building the Bridge Between Business and Technology

The most profound transformation has been the development of a robust architectural mindset. Before this course, my approach to system design was fragmented, focusing on individual components rather than their interconnections and business value. What I particularly appreciated about the ArchiMate training was how it provided a common language that everyone - from business stakeholders to technical teams - could understand and use effectively. Through rigorous practice with ArchiMate, I learned to construct coherent, multi-layered service blueprints that clearly articulate how business objectives drive application components, which in turn are supported by technology infrastructure. This holistic perspective enables me to visualize complex systems in their entirety while understanding intricate dependencies between layers.

For instance, I can now model how a business process like "customer onboarding" is supported by specific application services (identity verification APIs, customer profile management), which then rely on cloud infrastructure services (serverless computing, database instances, API gateways). This capability to create precise visual representations has proven invaluable in bridging the communication gap between business stakeholders and technical teams. It allows me to translate abstract business requirements into concrete technical specifications and, conversely, explain technical constraints and opportunities in business terms that executives can understand and appreciate. The course's emphasis on standardized notation and comprehensive viewpoint development has made me much more effective in technical communication and consensus-building.

2. Service Lifecycle Management and Operational Awareness: Designing for Sustainability

The course instilled in me a profound appreciation for the entire service lifecycle—from conception to retirement. What stood out to me was how the course connected theoretical lifecycle concepts to real-world operational challenges. Understanding the four major lifecycles (Application Service, Infrastructure Service, IT Asset, and Technology Product) has fundamentally changed my approach to architecture design. I no longer focus solely on initial design and deployment but consider the long-term sustainability and evolution of services. This operational awareness, heavily emphasized in the ITIL modules, means I now automatically consider the operational implications of every architectural decision.

When designing cloud architecture, I systematically think about how it will be monitored, updated, scaled, and eventually decommissioned. This includes considering aspects like patch management strategies, capacity planning, disaster recovery procedures, and technical debt management. The ITIL service lifecycle approach—particularly the transition from strategy to design, transition, operation, and continual improvement—has provided me with a structured framework to ensure services remain valuable and viable throughout their existence. The course's practical exercises around lifecycle management helped me understand that good architecture isn't just about technical excellence, but about creating solutions that can evolve and adapt over time.

3. Value-Oriented Design and Quantitative Evaluation: Measuring What Truly Matters

A crucial paradigm shift was internalizing the principle that technology's worth is determined solely by the business value it enables. The concepts of Utility (fitness for purpose) and Warranty (fitness for use) from ITIL became central to my evaluation criteria. I found the course's treatment of value measurement particularly insightful because it provided concrete tools and frameworks that I could immediately apply in real scenarios. I learned to move beyond traditional technical metrics like uptime and latency to define success in terms of tangible user outcomes and business goals. This involves using sophisticated measurement frameworks including Service Level Objectives (SLOs), Key Performance Indicators (KPIs), and Critical Success Factors (CSFs).

Furthermore, the Continual Service Improvement (CSI) model, particularly the Deming Cycle (Plan-Do-Check-Act) and the Seven-Step Improvement Process, provided a structured methodology for using these measurements to drive ongoing optimization. I can now proactively design services with built-in metrics collection, monitoring capabilities, and feedback loops to demonstrate their ROI and guide iterative improvements. This ensures services remain aligned with evolving business needs and can adapt to changing market conditions. The course's emphasis on measurable outcomes has made me more disciplined in defining success criteria and tracking progress against business objectives.

4. Framework Integration and Strategic Application: Mastering the Enterprise Toolkit

Initially, frameworks like TOGAF, COBIT, and ITIL appeared as separate, siloed bodies of knowledge. What made this course exceptional was how it demonstrated the practical integration of these frameworks in real enterprise contexts. The course enabled me to understand their powerful synergies and how to integrate them strategically within enterprise contexts. I now perceive them not as competing standards but as complementary tools in an architect's toolkit, each serving specific purposes while working together harmoniously.

For example, in a cloud transformation initiative:

I would employ TOGAF's Architecture Development Method (ADM) as the overarching governance-heavy approach to plan and execute the transformation strategy

I would utilize ArchiMate within the TOGAF process to create precise models for Baseline, Transition, and Target Architectures

I would leverage COBIT's governance objectives and management practices to ensure the cloud environment remains risk-aware, compliant, and delivers measurable value

I would implement ITIL's practices around Service Design, Transition, and Operation to ensure day-to-day delivery and support of cloud services are efficient and user-focused

This integrated understanding allows me to select and combine the most relevant aspects of these frameworks to address specific organizational challenges, rather than attempting to adopt any single framework rigidly. The course's practical approach to framework integration has given me confidence in selecting and adapting the right tools for each unique situation.

This integrated understanding has crystallized in my mind as a powerful conceptual model—one where TOGAF, COBIT, and ITIL function not as separate entities but as interlocking gears within a cohesive digital transformation engine. In this mental framework, TOGAF serves as the central drivetrain, providing the methodological structure for the entire journey from vision to implementation. COBIT acts as the governance system, ensuring the engine stays on track, manages risks, and delivers measurable value. Meanwhile, ITIL operates as the finely tuned operational core, guaranteeing the efficiency and reliability of delivered services. ArchiMate, in turn, provides the universal blueprint that enables clear communication and alignment across all components. This conceptual model has proven invaluable in practice, such as when planning cloud migrations where TOGAF charts the course, COBIT ensures governance, ITIL guarantees operational excellence, and ArchiMate creates the shared understanding that keeps all stakeholders aligned.

5. Critical Analysis and Future Development: Extending Course Concepts to Emerging Technologies

The INFS 604 course has not only provided me with fundamental knowledge but has also equipped me with the analytical framework to evaluate and adapt to emerging technologies. Through the systematic approach taught in the course, I can now effectively assess how new technological trends like edge computing, AI-powered operations, and serverless architectures fit into existing enterprise frameworks. The course's emphasis on structured evaluation methods has enabled me to develop a critical perspective that balances innovation with practical implementation considerations.

What sets this learning apart is the way the course connected traditional enterprise architecture concepts with modern cloud-native paradigms. For instance, the ITIL service value chain concepts have proven equally applicable to containerized microservices architectures as they were to traditional monolithic systems. Similarly, the ArchiMate modelling skills have allowed me to effectively represent complex hybrid cloud environments that span multiple cloud providers and on-premises infrastructure. This ability to adapt established frameworks to new technological contexts has been one of the most valuable outcomes of the course.

Looking forward, I recognize that my architectural journey requires continuous learning. While the course has provided a solid foundation in core frameworks, I plan to deepen my expertise in cloud-specific architectures and emerging paradigms. The concepts learned in INFS 604 will serve as my guiding principles as I explore more advanced topics in cloud economics,

multi-cloud governance, and sustainable architecture practices. This course has fundamentally shaped how I approach technology decisions - always considering the broader business context, long-term implications, and value delivery rather than just technical features.

Bridging Theory and Practice: The Huawei Cloud Experience

My internship at Huawei Cloud provided a real-world environment where these academic concepts came to life. What made this theoretical-practical connection so powerful was seeing how the course concepts scaled in enterprise environments. Observing the internal service catalogs, SLA negotiation processes, and operational monitoring systems offered practical demonstrations of ITIL and COBIT principles in action. This experience solidified my understanding of how abstract frameworks manifest in the operations of a leading cloud service provider. The course provided the theoretical foundation and vocabulary to deconstruct and understand why processes were established in particular ways, transforming me from a passive observer to an analytically engaged professional.

Personal Reflection on the Course

What I found most valuable about INFS 604 was its balanced approach between theoretical depth and practical application. The course didn't just teach frameworks and concepts - it showed us how to apply them in realistic scenarios through case studies and hands-on exercises. The emphasis on architectural thinking rather than just technical implementation has permanently changed how I approach problems. I particularly appreciated how the course connected different frameworks together, showing how they complement rather than compete with each other.

However, I believe the course could be enhanced by including more case studies from different industries and organizational sizes. While the concepts are universally applicable, seeing how they play out in startups versus large enterprises would provide additional valuable perspectives. Additionally, more guidance on navigating organizational politics and change management when implementing architectural transformations would be beneficial, as these are often the biggest challenges in real-world scenarios.

In conclusion, INFS 604 has equipped me with more than just knowledge—it has forged a new professional identity. I have transitioned from focusing primarily on technical implementation—the "how to build"—to embracing architectural thinking—the "what to build, why to build it, and how to ensure it delivers continuous value." I now possess the confidence to deconstruct complex business problems, design future-state service architectures, and create comprehensive roadmaps for their realization and governance. However, I recognize that true mastery requires ongoing practice and experience, particularly in applying the nuanced aspects of ITIL v4 practices and executing the detailed phases of the TOGAF ADM in large-scale environments. This course has provided the foundational map and compass for my career as a cloud architect, while the journey toward expertise continues through practical application and continued learning. The course has not only given me valuable skills but has also inspired me to continue learning and growing as a professional in the ever-evolving field of cloud architecture.