

COURSEWORK PART A+B

First Name - **JOHN-ALEX**

Surname - **YANNOULIAS**

Student ID - **W1775197**

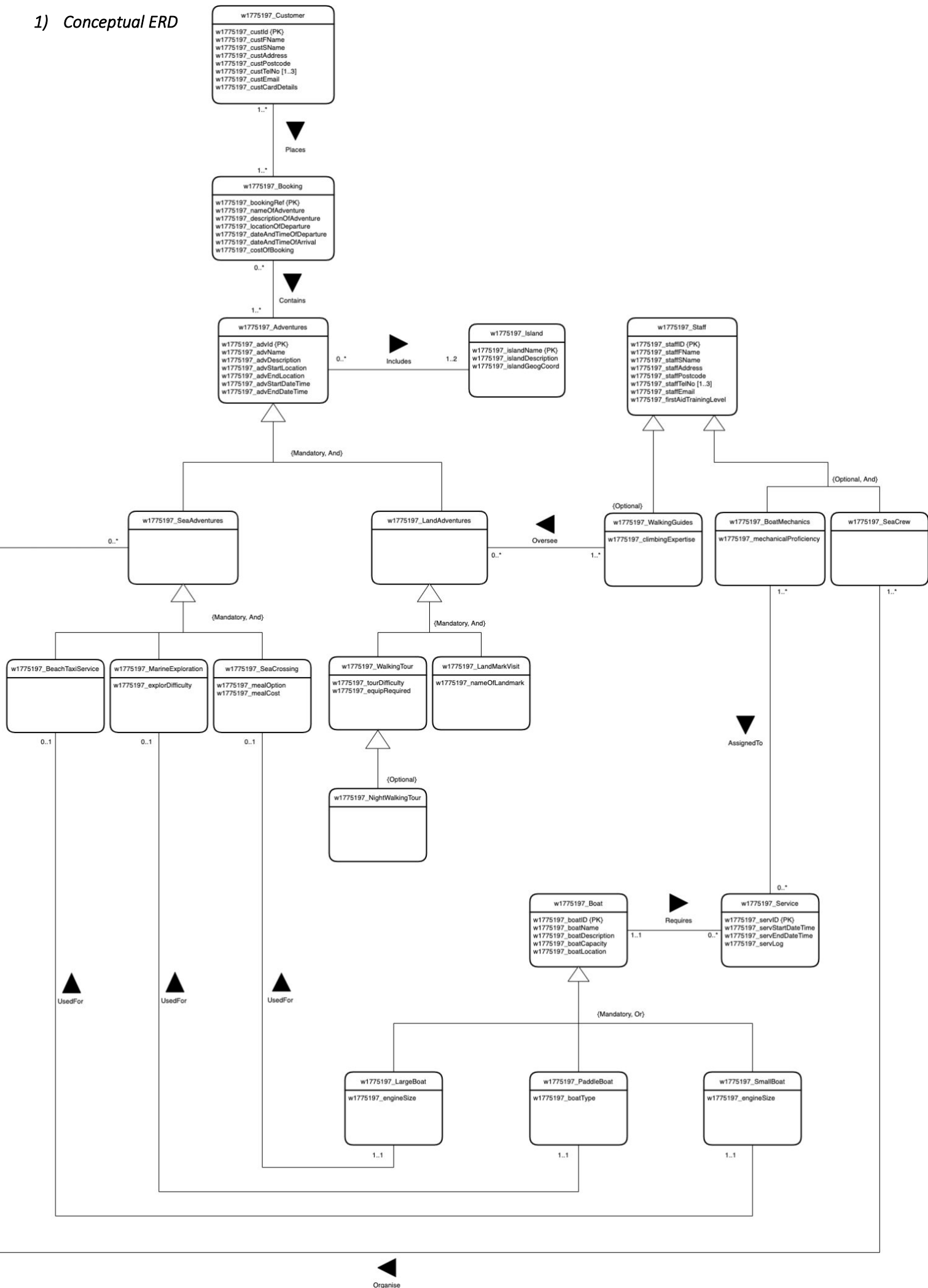
Tutorial Date & Time – **Thursdays 14:00-16:00 G.100**

5COSC020W Database Systems

Tutorial Group - **5SE01**

CONTENTS	
Conceptual ERD (Part A)	Page 2
Data Dictionaries (Part A)	Page 3-9
Logical ERD (Part B)	Page 10
SQL Query (Part B)	Page 11
Comparative Analysis Table (Part B)	Page 12-13

1) Conceptual ERD



2) Data Dictionaries – Entities

Entity Name	Brief Description
w1775197_Customer	General term describing all Customers of ArchipelagoCrazy
w1775197_Booking	General term describing all Bookings of ArchipelagoCrazy
w1775197_Adventures	General term describing all Adventures offered by ArchipelagoCrazy
w1775197_Island	General term describing the collection of Islands where ArchipelagoCrazy operates
w1775197_Staff	General term describing all Staff employed by ArchipelagoCrazy
w1775197_Service	General term for all the types of Servicing that ArchipelagoCrazy carries out
w1775197_Boat	General term for all the Boats owned by ArchipelagoCrazy

General Entity	Specialised Entity	Brief Explanation
w1775197_Adventures	w1775197_SeaAdventures	Sea Adventures is a type of Adventure that offers customers various activities involving the Sea, Beaches, Islands etc.
w1775197_Adventures	w1775197_LandAdventures	Land Adventures is a type of Adventure that offers customers various activities involving the towns, villages, landmarks etc. on the collection of Islands.
w1775197_SeaAdventures	w1775197_BeachTaxiService	A Beach Taxi Service is a type of Sea Adventure that involves travelling on a Small Boat to a Beach or between Beaches.
w1775197_SeaAdventures	w1775197_MarineExploration	A Marine Exploration is a type of Sea Adventure that involves using a paddle propelled boat (kayak, canoe etc.) to explore an island locally and access the more remote bays and creeks.
w1775197_SeaAdventures	w1775197_SeaCrossing	A Sea Crossing is a type of Sea Adventure that involves using a Large Boat to carry people from the mainland to the islands but also between different islands.
w1775197_LandAdventures	w1775197_WalkingTour	A Walking Tour is a type of Land Adventure that involves being accompanied with a guide(s) to view an Islands physical on land beauty.
w1775197_LandAdventures	w1775197_LandMarkVisit	A Landmark Visit is a type of Land Adventure that involves being accompanied with a guide(s) to view key landmarks on the collection of Islands such as castles, fortresses, lighthouses, churches, monasteries, etc.
w1775197_WalkingTour	w1775197_NightWalkingTour	A Night Walking Tour is a type of Walking Tour that involves being accompanied with a guide(s) to view an Islands physical on land beauty during the Night.
w1775197_Staff	w1775197_WalkingGuides	A Walking Guide is a type of Staff that oversees both the types of Land Adventures, takes customers out on walks & takes customers on cultural visits.

w1775197_Staff	w1775197_BoatMechanics	A Boat Mechanic is a type of Staff that ensures the maintenance of all the Boats by servicing them.
w1775197_Staff	w1775197_SeaCrew	A Sea Crew is a type of Staff that helps organise and manage all the 3 types of Sea Adventures.
w1775197_Boat	w1775197_LargeBoat	A Large Boat is a type of Boat that can handle a higher number of customers and used for crossings to carry people from the mainland to the islands but also between different islands
w1775197_Boat	w1775197_PaddleBoat	A Paddle Boat is a type of Boat that can be used to explore an island locally and access the more remote bays and creeks.
w1775197_Boat	w1775197_SmallBoat	A Small Boat is a type of Boat that can take 6-8 people. They are used to provide a local beach taxi service on the larger islands to allow Customers to conveniently travel to the main beautiful beaches.

3) Data Dictionaries – Relationships & Multiplicities

Entity Name	Multiplicity	Relationship	Multiplicity	Entity Name	Brief justifications for the multiplicity
w1775197_Customer	1..*	Places	1..*	w1775197_Booking	One booking contains at least 1 customer – you can't have a booking without a customer.
					One booking could contain many customers – MD has stated a booking could contain many customers.
					One Customer must place at least 1 booking – MD stated a customer must have a booking in order to be on the system.
					One Customer can place many bookings – MD stated a customer can create as many separate bookings under the same name as they desire.
w1775197_Booking	0..*	Contains	1..*	w1775197_Adventures	One adventure may be assigned to 0 bookings – Perhaps no one has booked this adventure yet (New Adventure, Quiet time of year etc.)
					One adventure may be assigned to many bookings – MD has stated many customers can book the same adventure.
					One Booking Contains at least 1 adventure – you can't book anything but adventures.
					One Booking may contain many adventures – MD had stated Customers can book as many adventures as they like.
w1775197_Adventures	0..*	Includes	1..2	w1775197_Island	One island may Include 0 Adventures – perhaps it is a quiet time of year.

					One island may be included in many Adventures – MD has stated many adventures can run on an island at once. One adventure includes at least 1 island – MD has stated all adventures include islands. One adventure at max can include 2 islands – the only adventure where multiple islands can be included are Sea crossings and Beach taxi service, where at max you travel from one island to another. Any further island travelling would be another adventure.
w1775197_WalkingGuides	1..*	Oversee	0..*	w1775197_LandAdventures	One Land adventure is assigned to at least 1 walking guide – MD stated all Land adventures have walking guides. One Land adventure may be assigned to many walking guides – MD stated that if the group of customers is big enough the company will send more walking guides. One walking guide may oversee 0 Land adventures – may have been just hired, may not have any work yet etc. One walking guide may oversee many Land Adventures – they could have multiple jobs per day.
w1775197_BoatMechanics	1..*	AssignedTo	0..*	w1775197_Service	One service must include at least 1 boat mechanic – you cannot have a service without a boat mechanic. One service may include many boat mechanics – may be a complex job requiring many mechanics. One Boat mechanic may be assigned 0 Services – may be newly hired, may be newly fired, may be no jobs yet etc. One Boat mechanic may be assigned many Services – may be many jobs that day for that mechanic.
w1775197_Boat	1..1	Requires	0..*	w1775197_Service	One service must include at least 1 boat – you cannot have a service without a boat. One service at max can include 1 boat – each service must be logged in per boat. One boat may require 0 service – may be a new boat or a boat that hasn't got any problems or needs of servicing. One boat may require many service – may be a boat with multiple types of servicing wrong with it (Engine service, Paint Service & all other types of required boat servicing)
w1775197_SeaCrew	1..*	Organise	0..*	w1775197_SeaAdventures	One sea adventure must include at least 1 sea crew – a business requirement stated by the MD.

					One sea adventure may include many sea crew – could be a bigger job requiring more staff. One sea crew may organise 0 sea adventures – newly hired, newly fired, may be no jobs yet etc. One sea crew may organise many sea adventures – may be many jobs that day for that that crew member.
w1775197_LargeBoat	1..1	UsedFor	0..1	w1775197_SeaCrossing	One Sea Crossing must include at least 1 Large Boat – A large boat is needed to cross the sea to the islands. One Sea Crossing can at max includes 1 Large Boat – I have asked the MD - “one sea crossing is only operated by one large boat. If a second boat is involved, then it is a different sea crossing.” One Large boat may be used for no Sea crossings – may be a quiet time of year, the boat may have recently broken down etc. One Large boat at max can be used for 1 Sea Crossing – at any one time the boat can only perform one crossing.
w1775197_PaddleBoat	1..1	UsedFor	0..1	w1775197_MarineExploration	One marine exploration must include 1 paddle boat – A paddle boat is needed in order to be a marine exploration. One marine exploration at max can include 1 paddle boat – if the customer wants more than one paddle boat, they must create another booking. One paddle boat may be used for 0 marine explorations – May have recently broken, may be a quiet time of year etc. One paddle boat can be used at max for 1 marine exploration – at any one time the boat can only perform one marine exploration.
w1775197_SmallBoat	1..1	UsedFor	0..1	w1775197_BeachTaxiService	One Beach taxi service must include 1 small boat – A small boat is needed in order to be a beach taxi service. One Beach taxi service at max includes 1 small boat – as stated by the MD if a second boat is used it is a different taxi service. One small boat may be used for 0 Beach taxi services – quiet time of year, newly broken boat. One small boat at max can be used for 1 Beach taxi service – at any one time the small boat can only perform one beach taxi service.

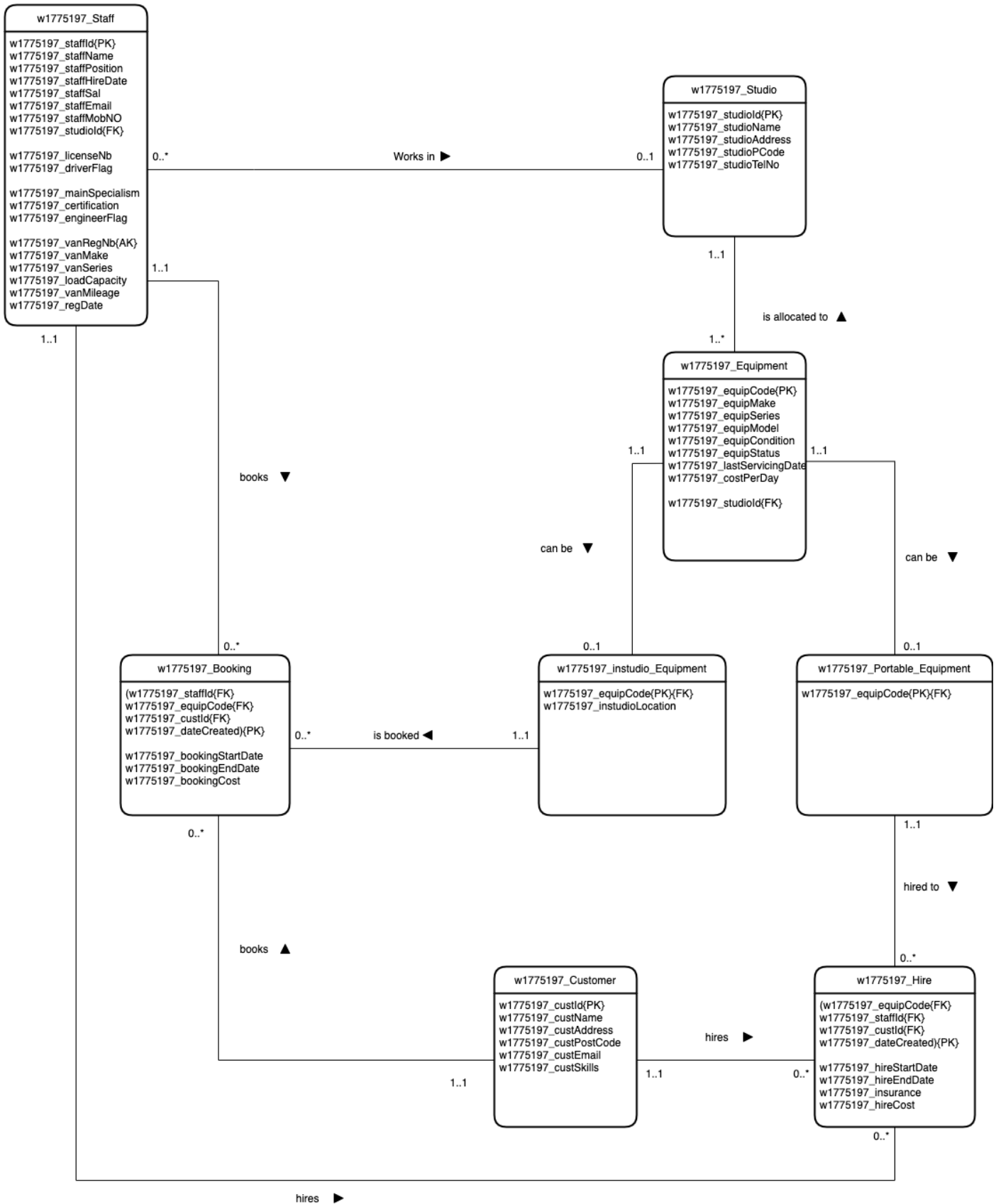
4) Data Dictionaries – Attributes & Primary Keys

Entity Name	Attributes for this entity (include PK)	Justification
w1775197_Customer	w1775197_custId {PK}	Uniquely identifies a customer of ArchipelagoCrazy
w1775197_Customer	w1775197_custFName	First Name of Customer
w1775197_Customer	w1775197_custSName	Second Name of Customer
w1775197_Customer	w1775197_custAddress	Address of Customer
w1775197_Customer	w1775197_custPostcode	Postcode of Customer
w1775197_Customer	w1775197_custTelNo [1..3]	Telephone number of Customer – can be up to 3 Values
w1775197_Customer	w1775197_custEmail	Email of Customer
w1775197_Customer	w1775197_custCardDetails	Card Details of Customer
w1775197_Booking	w1775197_bookingRef {PK}	Uniquely identifies a Booking of ArchipelagoCrazy
w1775197_Booking	w1775197_nameOfAdventure	Name of Adventure Booked
w1775197_Booking	w1775197_descriptionOfAdventure	Brief Description of Adventure
w1775197_Booking	w1775197_locationOfDeparture	Location of the departure of the booking
w1775197_Booking	w1775197_dateAndTimeOfDeparture	Date and Time of departure of the booking
w1775197_Booking	w1775197_dateAndTimeOfArrival	Date and Time of arrival of the booking
w1775197_Booking	w1775197_costOfBooking	Total Cost of the booking
w1775197_Adventures	w1775197_advId {PK}	Uniquely identifies an adventure created by ArchipelagoCrazy
w1775197_Adventures	w1775197_advName	Name of Adventure
w1775197_Adventures	w1775197_advDescription	Brief Description of Adventure
w1775197_Adventures	w1775197_advStartLocation	The starting location of the adventure
w1775197_Adventures	w1775197_advEndLocation	The end location of the adventure
w1775197_Adventures	w1775197_advStartDateTime	The start date and time of the adventure
w1775197_Adventures	w1775197_advEndDateTime	The end date and time of the adventure
w1775197_Island	w1775197_islandName {PK}	Uniquely identifies the Island by name
w1775197_Island	w1775197_islandDescription	Brief description of the island
w1775197_Island	w1775197_islandGeogCoord	The geographical coordinates of the island

w1775197_LandMarkVisit	w1775197_nameOfLandmark	The name of the landmark visited
w1775197_Staff	w1775197_staffID {PK}	Uniquely identifies the staff members of archipelagoCrazy
w1775197_Staff	w1775197_staffFName	Staff First name
w1775197_Staff	w1775197_staffSName	Staff second name
w1775197_Staff	w1775197_staffAddress	Staff address
w1775197_Staff	w1775197_staffPostcode	Staff postcode
w1775197_Staff	w1775197_staffTelNo [1..3]	Staff telephone numbers – could be up to values
w1775197_Staff	w1775197_staffEmail	Staff email
w1775197_Staff	w1775197_firstAidTrainingLevel	The level of first aid training qualification obtained by the staff
w1775197_WalkingGuides	w1775197_climbingExpertise	The level of climbing expertise obtained by the staff
w1775197_BoatMechanics	w1775197_mechanicalProficiency	The level of mechanical expertise obtained by the staff
w1775197_Service	w1775197_servID {PK}	Uniquely Identifies the service by using an ID
w1775197_Service	w1775197_servStartDateTime	The Service start date and time
w1775197_Service	w1775197_servEndDateTime	The Service end date and time
w1775197_Service	w1775197_servLog	A Service log to keep log of all the services for ArchipelagoCrazy to keep on top of safety
w1775197_Boat	w1775197_boatID {PK}	Uniquely Identifies boat by using an ID
w1775197_Boat	w1775197_boatName	Name of the boat
w1775197_Boat	w1775197_boatDescription	Brief description of the boat
w1775197_Boat	w1775197_boatCapacity	Capacity of the boat
w1775197_Boat	w1775197_boatLocation	Location of the boat
w1775197_LargeBoat	w1775197_engineSize	Engine size of the large boat
w1775197_SmallBoat	w1775197_engineSize	Engine size of the small boat
w1775197_PaddleBoat	w1775197_boatType	Type of paddle boat (canoe, kayak etc.)
w1775197_MarineExploration	w1775197_explorDifficulty	Difficulty level of the exploration – business requirement

w1775197_SeaCrossing	w1775197_mealOption	Meal option of the sea crossing – business requirement
w1775197_SeaCrossing	w1775197_mealCost	The cost of the meal on board the sea crossing – business requirement
w1775197_WalkingTour	w1775197_tourDifficulty	Difficulty level of the Walking tour – business requirement
w1775197_WalkingTour	w1775197_equipRequired	Equipment required for the walking tour – business requirement

5) Logical ERD



6) SQL Query

```
SELECT
    w1775197_Studio.w1775197_studioName,
    w1775197_Equipment.w1775197_equipMake,
    w1775197_Equipment.w1775197_equipSeries,
    w1775197_Equipment.w1775197_equipModel
FROM
    w1775197_Studio JOIN w1775197_Equipment
ON
    w1775197_Studio.w1775197_studiold = w1775197_Equipment.w1775197_studiold
WHERE
    w1775197_Studio.w1775197_studioAddress LIKE '%London%'
AND
    w1775197_Equipment.w1775197_costPerDay > 125;
```

7) Comparative Analysis Table

	Relational Databases (SQL)	Non-Relational Databases (NoSQL)
Model / Storage	Relational data is stored in a table-based structure (Jatana et al., 2012).	Non-Relational data is stored in documents (JSON), key-value pairs, XML format and graph databases (Jatana et al., 2012).
History	Developed in 1970 by E.F Codd (Jatana et al., 2012).	First used in 1998 for a relational database that omitted the use of SQL (Strauch, 2011), thus it is still a new technology however gaining a lot of popularity.
Schemas / Infrastructures	In a relational database, it is required to define your schema (Wang and Yang, 2017). This is formally known as a predefined table-based schema. Altering the schema in a relational database can be very costly both in time and money, often involve downtime and would be disruptive (Smallcombe, 2021).	In a non-relational database, a dynamic schema is utilised, which allows the use of what's known as "unstructured data" (Wang and Yang, 2017). Meaning the application can be built first without having to predefine the schema. Not having to first define the schema makes NoSQL databases much easier to update as data changes (Jatana et al., 2012).
Language	Uses Structured Query Language (SQL) to define and manipulate data (Jatana et al., 2012).	There isn't a set language – different NoSQL databases use a different syntax (Jatana et al., 2012).
Data Consistency	All SQL databases follow a strong consistency model - (ACID - Atomic, Consistent, Isolation and Durable) ensures that once a transaction is complete its data is stable and consistent with the disk (Jatana et al., 2012).	Most NoSQL Databases follow an eventual consistency model (BASE - Basic Availability, Soft-State and Eventual Consistency). BASE is a lot less strict than ACID. Data in the NoSQL Databases will be consistent in the future / eventually. The BASE model prefers availability rather than the consistency of data. (Jatana et al., 2012).
Performance	Additional security measures such as encrypted messages, role-based security etc. leads to slower performance (Jatana et al., 2012).	NoSQL is horizontally scalable leading to high performance. NoSQL read and write performance is much faster than relational databases (Jatana et al., 2012).
Workload	Relational Databases handle complex queries very well as SQL has a standard interface for handling queries (Jatana et al., 2012). NoSQL lacks the ability to perform dynamic querying.	Better for modern applications that have complex constantly changing data sets, requiring a flexible data model that doesn't need to be immediately defined (MongoDB, no date). The Agile and flexible features as well as the ability of storing and processing data in real time, attracts a lot of developers and organizations (MongoDB, no date).
Security	Strong security features built-in. Secure database transactions are ensured by following the ACID rules (MongoDB, no date).	To allow for faster data access, very few security features are built-in. More prone to security attack as doesn't follow ACID rules (MongoDB, no date).
Scaling	SQL databases are vertically scalable (MongoDB, no date). As the entire database is hosted on a single server, scaling the server involves adding more CPU, RAM, or SSD capacity (buying a bigger, more expensive server) (Smallcombe, 2021). This is usually very expensive.	NoSQL databases are horizontally scalable (MongoDB, no date). This involves handling higher traffic by sharding, or adding more commodity servers in the NoSQL database. Compared to Vertical scaling, Horizontal scaling has a greater overall capacity, making NoSQL databases the ideal choice for enormous and ever-changing data sets (Smallcombe, 2021). This is also far cheaper compared to vertical scaling (MongoDB, no date).

Bibliography –

Jatana et al., (2012). A survey and comparison of relational and non-relational database. *International Journal of Engineering Research & Technology*, 1 (6).

MongoDB. (no date). *NoSQL vs Relational Databases*. [online] Available from <https://www.mongodb.com/scale/nosql-vs-relational-databases> [Accessed 2 November 2021].

MongoDB. (no date). *NoSQL vs SQL Databases*. [online] Available from <https://www.mongodb.com/nosql-explained/nosql-vs-sql> [Accessed 2 November 2021].

Smallcombe, M., (2021). SQL vs NoSQL: 5 Critical Differences. [Blog] *Xplenty*, Available from <https://www.xplenty.com/blog/the-sql-vs-nosql-difference/> [Accessed 1 November 2021].

Strauch, C., Sites, U.L.S. and Kriha, W., (2011). NoSQL databases [*Lecture Notes*]. *Stuttgart Media University*. Available from https://www.researchgate.net/profile/Jesus-Sanchez-Cuadrado/publication/257491810_A_repository_for_scalable_model_management/links/568baf0508ae051f9afc5857/A-repository-for-scalable-model-management.pdf [Accessed 1 November 2021].

Wang, R. and Yang, Z., (2017). *SQL vs NoSQL: A Performance Comparison*. New York : University of Rochester. Available from <https://www.cs.rochester.edu/courses/261/fall2017/termpaper/submissions/06/Paper.pdf>> [Accessed 1 November 2021].