**GPS Address**

* Normalised GPS into it's own entity
* Address entity for customers who don't need GPS
* GPS\_address entity for e.g. hotels/resorts/airports which will have GPS address which wraps address\_id and gps\_id
* Latitude  GPSdec(8,6) because it requires 8 digits in total 6 of which  are precision. Max/min latitude GPS values  is –90 to +90
* Longitude dec(9,6) which is 90 digits total, 6 for precision. Max/min values are –180 - +180

**Booking Duration**

* Encoded by flight/date times and would be identified via SQL query

**Room Type vs Room (Hotel ID)**

* Linked hotel\_id to room\_type\_id so each hotel has it's own room type data

Nothing external to jet2Holidays (no portals)

Luggage, baggage, travel insurance, transfers all deemed out of scope and upsells

Jet2Holidays is a package provider thus flight only/ holiday only options are deemed out of scope

Multiple destinations for one holiday booking is deemed out of scope

All flights will be direct flights

Query around reviews

Jet2 uses 3rd party (Trip Advisor) for review info. Does this mean it's out of scope for database?

**GPS Assumptions**

GPS latitudes data type are decimal(8,6) requires 8 digits total, 6 of which are decimal point precision (latitude range –90 to 90

GPS longitudes data type are decimal (9,6) (longitude range –360 to 360)

**Log-In**

Website doesn't appear to have username & password log-in

Instead has booking reference, lead passenger surname & DOB

**Currency**

Currency always appears to be in GBP regardless of where website accessed

Therefore will store pricing at all times in GBP and front end developer will apply currency exchange at run

Table

Description automatically generated

**Passenger Table**

* Title normalised to avoid redundancy in data
* DOB will be used to derive age (rather than it's own attribute)
* Passport\_id also normalised as per 1NF

Graphical user interface, application, table

Description automatically generated

**Currency & Language**

* Included tables as future-proofing incase company decides to support bookings from non-UK destinations
* Increase lifespan of design (Chen)

Chart

Description automatically generated with medium confidence

**Room, room\_price & room type**

**Room**

* Room type normalised

Base quantity in room type

* + Availability model- determined by base\_quantity leveraged against bookings of that room type in a given date range

**Room Price**

* Design decision to set price at room\_type level for given time periods
* Valid from and to gives flexibile, dynamic pricing
* Intention is for front end to update pricing depending on availability via SQL transaction (room booking dates vs base quantity)

**General Pricing Assumption**

* **Pricing incorporating off/on peak:**
* Likely need to have dynamic pricing for dates both for hotel room booking and flights
  + Date A – Date B -> Flights Price A
  + Date C – Date D -> Flights Price B
    - Same for hotel room booking
* Further investigation as this could make system unwieldly (poss flat pricing for flights and discussion point around "future improvements

**Question for Neil**

* Is board type pricing considered upselling and therefore out of scope?

Graphical user interface, application

Description automatically generated