SIBD Project - Part 1

Emy Marie Bimond
ist1116694

Ilia Golub
ist1118697

Ovidiu-Gabriel Lasconi
ist1116578

Group: 20

Laboratory teacher: Professor Paulo Carreira, Professor Francisco Regateiro

Contributions and effort: All three group members contributed approximately equally to the project, each with about one third (~33%) of the work and roughly 6 hours of effort.

Our data model (Figure 1) for the Boating Management System is organized into five areas: boats and classes, sailors and responsibilities, reservations and trips, geography and jurisdictions, and certifications. This mirrors the problem description while keeping the E-A diagram readable.

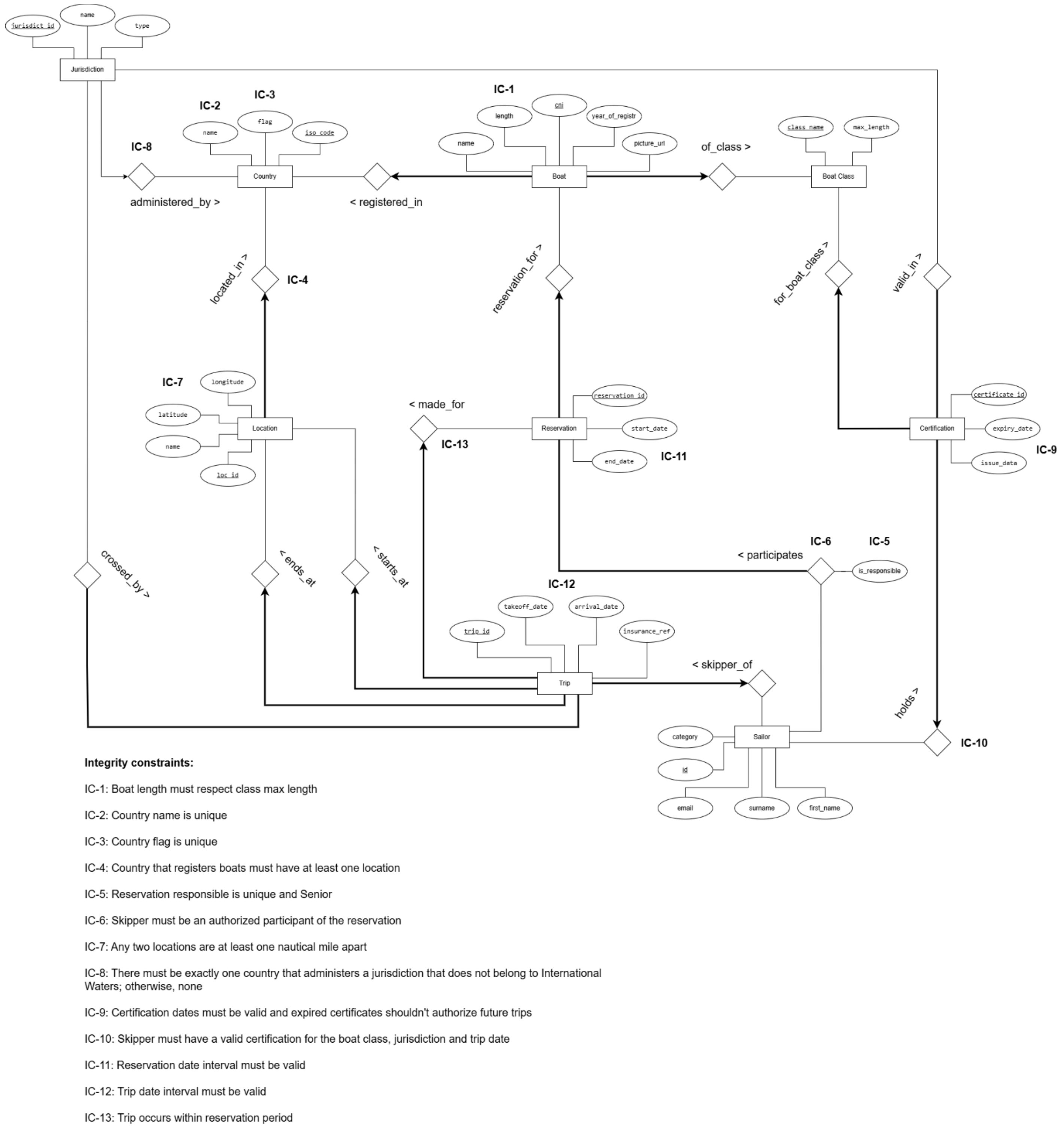
Boats are represented by entity **BOAT**, with attributes such as the national identifier cni, name, length, year_of_registr and picture_url. Each boat belongs to exactly one **BOAT_CLASS** through relationship OF_CLASS; **BOAT_CLASS** stores class_name and max_length. This supports the constraint that a boat's length must not exceed the maximum for its class. Boats are registered in exactly one **COUNTRY**, represented by attribute flag_iso_code referencing COUNTRY(iso_code). **COUNTRY** also stores a unique name and flag, together with a constraint that any country that registers boats must have at least one **LOCATION**.

Sailors are modeled as a single entity **SAILOR** with attribute category ∈ {"Senior", "Junior"}. We avoid an hierarchy because the two kinds of sailors share the same attributes and differ only in their role in reservations and trips. Using one entity with a constrained attribute yields the same semantics with a simpler model. The many-to-many relationship **PARTICIPATES** between **RESERVATION** and **SAILOR** represents the authorized sailors of a reservation and has attribute is_responsible. A constraint states that each reservation must have at least one participant, exactly one participant with is_responsible = TRUE, and that this sailor must be Senior. Modeling responsibility as an attribute on **PARTICIPATES** avoids a separate "responsible" relationship and an extra constraint to ensure the responsible sailor is also a participant.

We distinguish clearly between **RESERVATION** and **TRIP**. A reservation links a single boat (boat_cni) to a time interval [start_date, end_date]. Trips are concrete journeys under a reservation, with their own dates, start and end locations and insurance_ref, and are linked to **RESERVATION** via a one-to-many relationship. **RESERVATION** and **TRIP** both have local checks on date order, and we add a constraint that every trip must lie inside the corresponding reservation interval. The relationship **SKIPPER** between **TRIP** and **SAILOR** identifies who operates the boat on that trip. Another constraint states that any skipper of a trip must also participate in the corresponding reservation, which captures that only authorized sailors may operate the boat.

Geographical and legal aspects are handled by **COUNTRY**, **LOCATION** and **JURISDICTION**. Locations have a name and coordinates and belong to exactly one country. Trips start and end at locations via attributes start_location and end_location referencing **LOCATION**. Jurisdictions represent legal zones and are implemented as an entity with attributes name and type, plus an optional iso_code referencing **COUNTRY**. A constraint on type and iso_code expresses the "International Waters" case: non-international jurisdictions must be administered by exactly one country, while International Waters have no country. Another constraint states that any two locations must be at least one nautical mile apart, using the latitude and longitude attributes.

Finally, we model **CERTIFICATE** as an explicit entity with attributes certificate_id, issue_date and expiry_date, and foreign keys to **SAILOR** and **BOAT_CLASS**. The association **VALID_IN** between **CERTIFICATE** and **JURISDICTION** captures the jurisdictions where a certificate is valid. This matches the requirement that a certification authorizes a sailor to act as skipper for a specific boat class in one or more country jurisdictions. A central constraint states that, for every trip, the skipper must hold at least one certificate whose class matches the boat's class, whose **VALID_IN** jurisdictions cover all jurisdictions crossed by the trip, and whose validity interval includes the trip's take-off date. This rule is too complex to implement with basic SQL constraints, but completes the link between certificates, trips, classes and jurisdictions.



Integrity constraints:

IC-1: Boat length must respect class max length

IC-2: Country name is unique

IC-3: Country flag is unique

IC-4: Country that registers boats must have at least one location

IC-5: Reservation responsible is unique and Senior

IC-6: Skipper must be an authorized participant of the reservation

IC-7: Any two locations are at least one nautical mile apart

IC-8: There must be exactly one country that administers a jurisdiction that does not belong to International Waters; otherwise, none

IC-9: Certification dates must be valid and expired certificates shouldn't authorize future trips

IC-10: Skipper must have a valid certification for the boat class, jurisdiction and trip date

IC-11: Reservation date interval must be valid

IC-12: Trip date interval must be valid

IC-13: Trip occurs within reservation period

Figure 1: Entity-Association diagram of the Boating Management System (boats, sailors, reservations, trips, geography, and certifications).