CS2102 Database Application Project
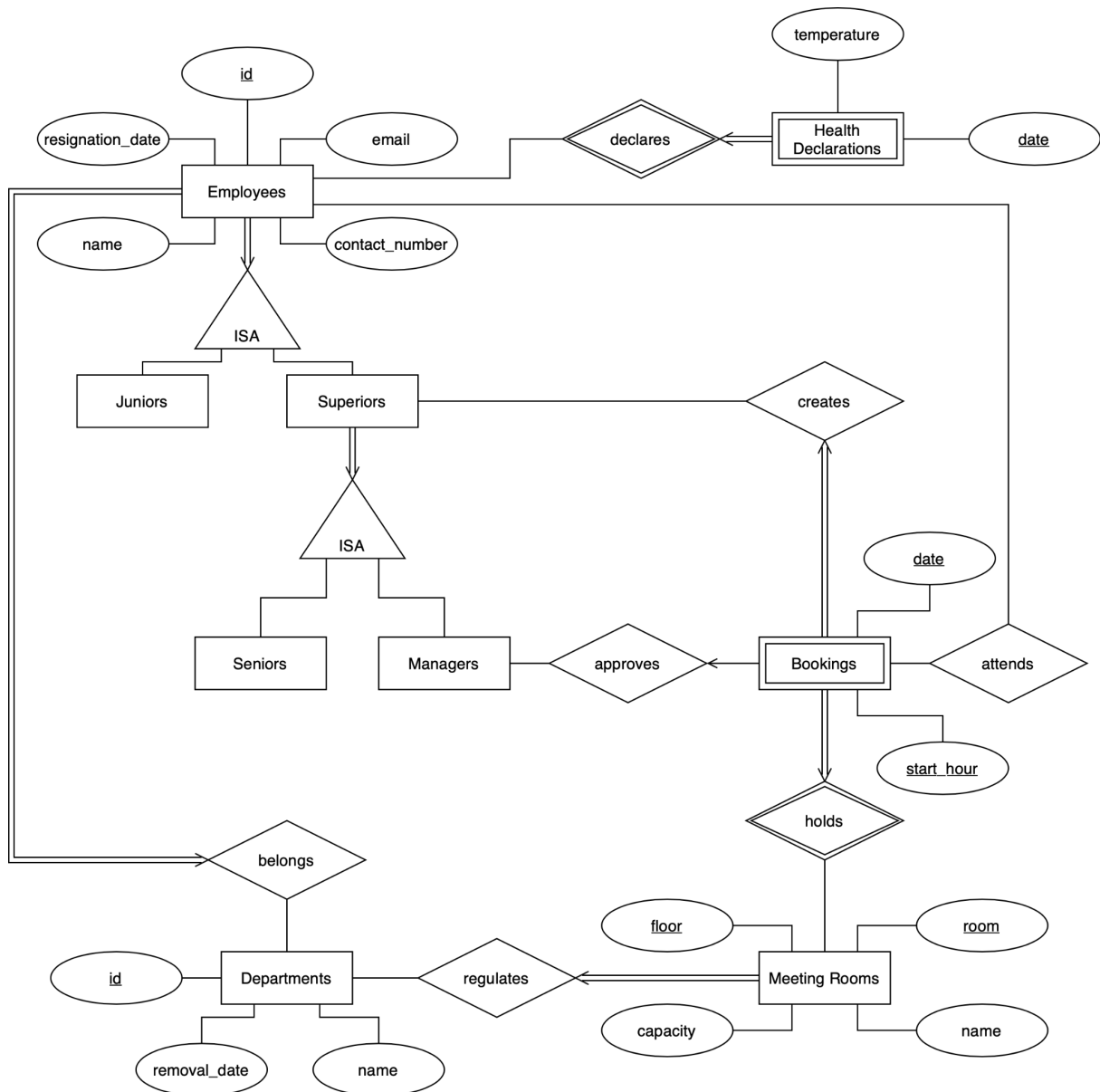# Entity-Relationship Data Model

Group 43

Members:
1. Goh Jun Wei
2. Tan Rui Xuan
3. Tan Ying Jie, Dexter
4. Wong Jun Long

ER Data Model

Justifications of Design Decisions

This section highlights the non-trivial design decisions made as well as their justifications.

1. <u>IS-A hierarchy between different types of employees</u>
   We used a IS-A relation hierarchy to represent the different seniority of employees, as compared to the alternative of using an attribute of the Employee entity. Furthermore, we have introduced an intermediate grouping, superiors, which comprises senior employees and managers. By using an IS-A hierarchy and explicitly separating manager entities from other types of employees, we are able to capture the constraint that only managers are able to approve bookings. In addition, with the introduction of the superiors grouping, we are able to capture the constraint in the ER data model that juniors are unable to create bookings and only seniors and managers are able to do so. The aforementioned constraints would not have been possible to be captured in an ER data model if the alternative design was chosen. We also note that the IS-A relation between all employee types are covering and do not allow for overlaps to indicate that an employee must be of a particular type.

2. <u>Dependency constraint between Health Declarations and Employees</u>
   We designed Health Declarations to be a weak entity set with Employees as its owner entity set. Furthermore, the date attribute forms the partial key of the Health Declarations entity set. This is in consideration that each health declaration record must be tied to a particular employee, and the date at which the health declaration was made uniquely identifies a daily health declaration for said employee.

3. <u>Dependency constraint between Bookings and Meeting Rooms</u>
   We designed Bookings to be a weak entity set with Meeting Rooms as its owner entity set. In addition, the partial key of Bookings comprises the date and the time of each booking. This is because each booking must be tied to a certain meeting room and each meeting room can only have a single booking given a specific date and time.

4. <u>Attributes of the Bookings entity set</u>
   Each booking entity has a start_hour and a date attribute, which together forms its partial key. In our chosen design, each booking lasts for exactly one hour and multi-hour meetings would be represented as multiple consecutive one-hour bookings. This design was chosen over the alternative where a multi-hour meeting would be represented as a single booking with the corresponding starting and ending hours as attributes.

   Our design choice has given us several benefits over the alternative. Firstly, we are able to capture the constraint that each meeting room can only hold a single meeting at a given point in time directly in the ER data model with the appropriate use of weak entity sets and careful selection of partial keys. Secondly, this implementation allows employees to join or exit multi-hour meetings at any of the hour marks as their attendance, through the attends relation, will be captured individually for each one-hour booking slot. Thirdly, managers are able to selectively approve parts of multi-hour meetings, with the restriction that these parts must start and end at hour marks.

   Additionally, we are also able to gain greater granularity in the contact tracing of close contacts as we can identify specific employees who were in the meeting room at the same time slots as an employee who has a fever.

As such, our chosen design allows for greater flexibility, and does not require any range checking to prevent conflicting bookings.

5. <u>Many-to-many relationship between Employees and Bookings</u>
   The relationship between Employees and Bookings is that of a many-many relationship because each employee can attend multiple bookings, albeit non-overlapping bookings, and each booking may be attended by multiple employees. As such, we have represented this association with the attends relationship.

6. <u>Approves relationship between Managers and Bookings</u>
   We have chosen to represent the approval of a booking by a manager as a many-to-one partial participation relationship. This relationship is named approves. Through the design, the presence of a manager-to-booking relationship indicates that the booking has been approved by the related manager and a lack of this relationship indicates that the booking is pending approval.

7. <u>Resignation status of an employee represented by resignation_date attribute</u>
   We have chosen to indicate if an employee has resigned through the use of a resignation_date attribute, where the presence of a value indicates that the employee has resigned effective from the date. This allows us to retain existing related records as the employee entity remains. In addition, using a date as compared to a boolean value allows us to easily determine related bookings, participations, or approvals which occur after the resignation date.

8. <u>Removal status of department represented by removal_date attribute</u>
   To represent the removal of a department, we have chosen to include an attribute, removal_date, which indicates the effective date from which a department was removed, and a lack of any value indicates that the department has not been removed. This design was chosen instead of deleting the removed department entity in order to preserve existing records such as meeting rooms that are regulated by the department, and the bookings that belong to these meeting rooms.

   [We note that while the application specifications do not explicitly state that a meeting room should be regulated by a specific department, the search_room routine requires each room to be associated with a department ID, and as such have inferred by extension that each meeting room must be associated with a department. We further note that due to a lack of a reassign_room routine to allow meeting rooms to be reassigned to other departments, the department entity cannot be deleted upon removal as rooms will need to be either left without a managing department or be deleted.]

Constraints Not Captured by ER Data Model
This section highlights the constraints of the application not captured by the ER data model.

1. Meeting room capacity constraints
   The ER data model does not capture the constraint where the total number of participants of a booking cannot exceed the room capacity, as well as the fact that the employee who created the booking is also counted as part of the participant list. Furthermore, the constraint where bookings which violate the maximum capacity of the meeting room after a capacity change are automatically removed is not captured.

2. Contact tracing related constraints
   The ER data model does not capture the constraint where employees with fever are unable to book meeting rooms, is removed from all future meeting rooms, and has his/her future bookings cancelled. In addition, it also does not capture the constraint where employees who were in close contact are also removed from their future meetings.

3. Constraints of resigned employees
   The ER data model does not capture the constraint where resigned employees cannot book or approve bookings, and that any future bookings, participation in bookings, or approvals are removed.

4. Constraints of approved meetings
   The ER data model does not capture the constraint where approved bookings can no longer have a change in participants.

5. Unique employee email constraint
   The ER data model does not capture the constraint where employees should have unique emails, in addition to unique IDs. This is because ER data models do not allow us to indicate the uniqueness of an attribute that is not a prime attribute.

[We also note that apart from the above mentioned constraints, the ER data model is also unable to capture some other constraints. Such as where only managers from the respective department in charge of the meeting room are able to change its capacity or approve any meetings that are held in said room, and where employees are unable to remove bookings of other employees.]