Playgrounds

Create a playground API (REST (Spring Boot) or library (jar) – which one is your decision) that implements business logic for playground management

These are things that we expect library will do:

- define clear and usable domain classes, beans, components (interfaces) that should be used access required functionality/data
- API creates and manages play sites that consists of double swings, carousel, slide and a ball pit. API allows to create combinations of play sites (for example two swings, one slide etc.)
- API should not use DB or persistent data store, In-memory storage is enough.
- API should be able to add kids to play sites (we know kid's name, age, ticket number)
- API should not allow to add more kids to them than specified in configuration of play site (done once on initialization)
- it should be possible to enqueue kid or receive negative result if kid does not accept waiting in queue. API register queues on play sites when tries to add kid to play site, that is full, and kid accepts waiting in queue).
- it should also be possible to remove kid from play site / queue
- API should provide play site utilization. Play site utilization is calculated differently for each play site (most of play sites utilization is percent of capacity taken (3/10 places taken for 30% utilization), double swings 100% if full capacity, 0 if 1 or 0 kids, some not yet known play sites can have different calculation implementations, this possible extension requirement should reflect in design). Utilization is measured in %
- API should be able to provide a total visitor count during a current day on all play sites

Please do not over-engineer, keep code simple and easy understandable.

If requirement or functionality is not clearly described please feel free implement it as you think would be logical. For example, "or receive negative result if kid does not accept waiting in queue " does not say how kid decides to not accept waiting so it is for you to decide the rule let's say 50/50. Similarly play site utilization calculation logic when play site has double swing together with other devices is up to you to decide.

For sake of simplicity do not take into account synchronization of possible parallel data access/modification.

Preferred shipment of homework is Github (or similar). Alternative could be zipped project sent via email.