TALLINK

Requirements

Test task should contain **README.MD** that will describe application startup process / data source credentials (in case if inside application use database users / etc.)

Test task should contain at least one integration test that will verify some processes logic for example: cancel conference when there is already added some participants.

Test task should contain at least unit test that will verify work of some utility classes logic / mappers / etc.

Test task should be stored on **GitHub** or **BitBucket** and link or invitation should be send to yury.repeshov@tallink.ee / villu.mets@tallink.ee.

After reading the task, please provide us your first time estimates for implementing the task and when you have time work on it.

TALLINK

Application architecture requirements

Programming languages (one of options)
- Java (version 18 or higher)
- Kotlin (latest available)

Data stores (one of options)
- MySQL 8.x or higher
- PostgreSQL 12.x or higher
- H2

Frameworks (one of options)
- Spring stack
- Hibernate / MyBatis / iBATIS
- Querydsl

Project builders (one of options)
- Maven 3.x
- Gradle 6.x

API development (one of options)
- OpenAPI 3.x (from Spring)

Database Migration
- Liquibase
- Flyway

Project containerization (optional)
- Docker for Kubernetes
- Docker Compose

TALLINK

Application functional requirements

As test task we expecting to see working conference management system, that will contain at least 2 modules: **Back Office Gateway** and **Conference Gateway**.

Back Office Gateway should support next functionality:

- Create new conference room(s)
  User should have option to create new rooms with multiple parameters: room name / room status / room location (aka physical address of place where this room located) / room max capacity

- Create new conference
  User should have option to create new conference in selected conference room at specified date and time. System also should before conference creation make multiple validations, like, is conference room is available at specified date and time / if conference room is available based on status (for example room status can be set UNDER_CONSTRUCTION / etc.)

- Cancel conference
  User should have option to cancel conference that is not yet happened. System also should cancel all registrations for conference that will be canceled.

- Check conference availability
  User should have option to check conference availability based on registered participants and conference room max capacity

- Update conference date time / room
  User should have option to update conference date time / update room where conference will be done

- Update conference room status / max capacity
  User should have option to update conference room status / max capacity. System should before update also apply such validations: if room status changed to UNDER_CONSTRUCTION and there are any conferences in future / if room max capacity is decreased and there is any conference where registered participants is more that new max capacity / etc.

- Find conference feedback(s)
  *User should have option to find feedback(s) from participants, where will be should feedback content and feedback author (NB! Due GDPR regulations system should show only participant first name and first letter or last name, like Yury R***********)*

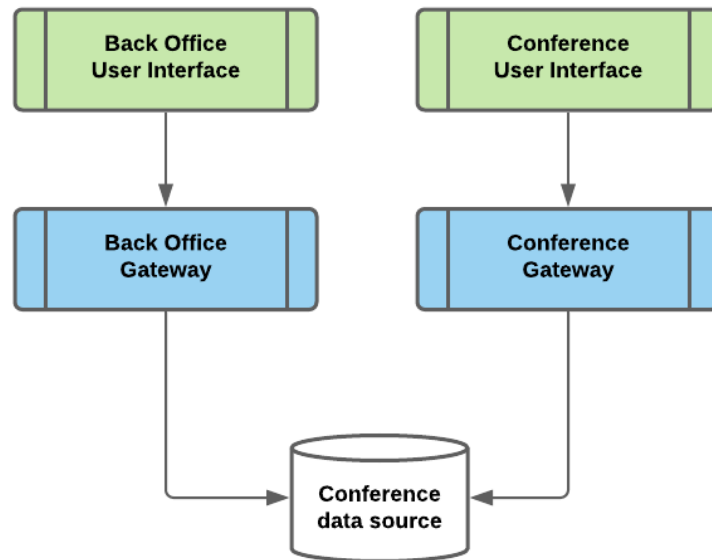**Conference Gateway** should support next functionality:

- Find available conference(s)
  User should have option to find available conferences for specified date time range and system should provide such information, like, amount of participates that already registered for this conference and where conference location

- Self-registration
  User should have option to registered itself to selected conference by filling such information like full name / gender / email / date of birth / etc. System should also validate such information like is conference room max capacity allow add new participant or not / is conference not canceled / etc. System should provide after successful registration unique conference participant code (NB! Internal id cannot be used, system should generate another unique code)

- Cancel registration
  User should have option using unique conference participant code cancel self-registration

- Send conference feedback
  User should have option to add feedback for conference using unique conference participant code.
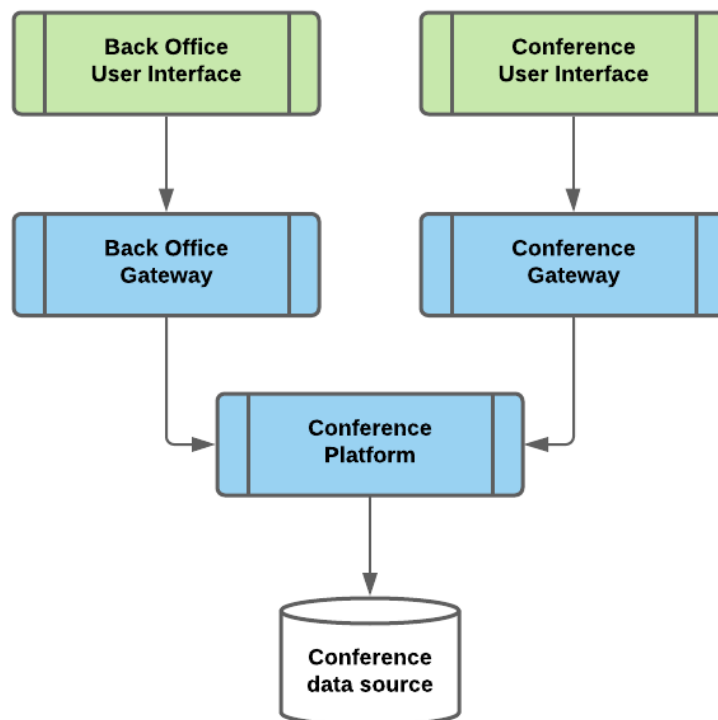
TALLINK

Application architecture example

Below can be found multiple application architecture examples that can be used during test task development. Also we always appreciate another application architectures that can used to solve test task.

Version 1

Version 2

Version 3