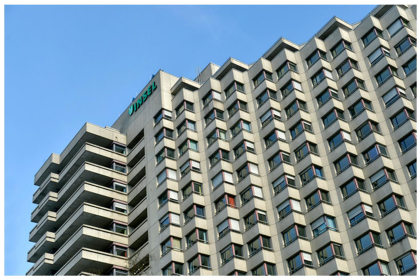
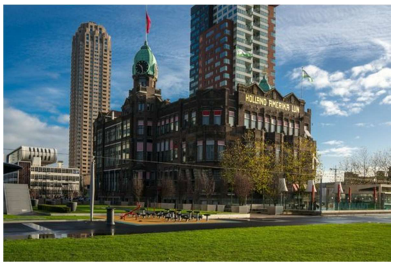
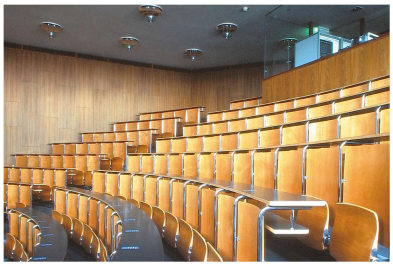
Documen-tation

Groupe Pink







Content

SRS-Project

Introduction

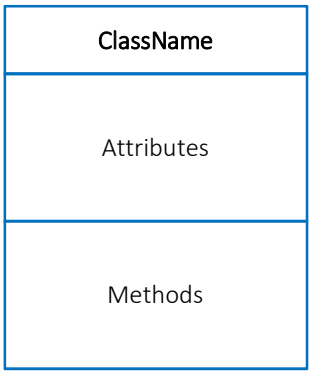
The goal of this project is to learn how to deal with development tools like EGit, Maven, ... and with development processes such as UML, Scrum, ... For this we have to design a *Smart Reservation System (SRS)*.
This program must be suitable for different types of enterprises. Be it sports clubs that rent out halls or hotels, which rent out meeting or overnight rooms, or a university, which wants to book teaching rooms for courses. The system shall allow its users to easily book any type of rooms.
The program has to run on different applications / devices. Users shall access the system via various kinds of applications such as standard Web interfaces and apps running on smartphones or tablets.

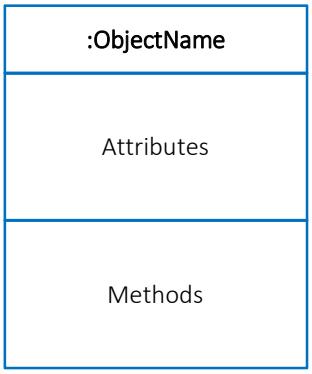
Requirements of the SR-System

- Reservations must be made easily
- Reservations can be simple time ranges or periodic events
- Time grids shall be made available in order to align reservations with time boundaries of, for example, given schedules.

UML

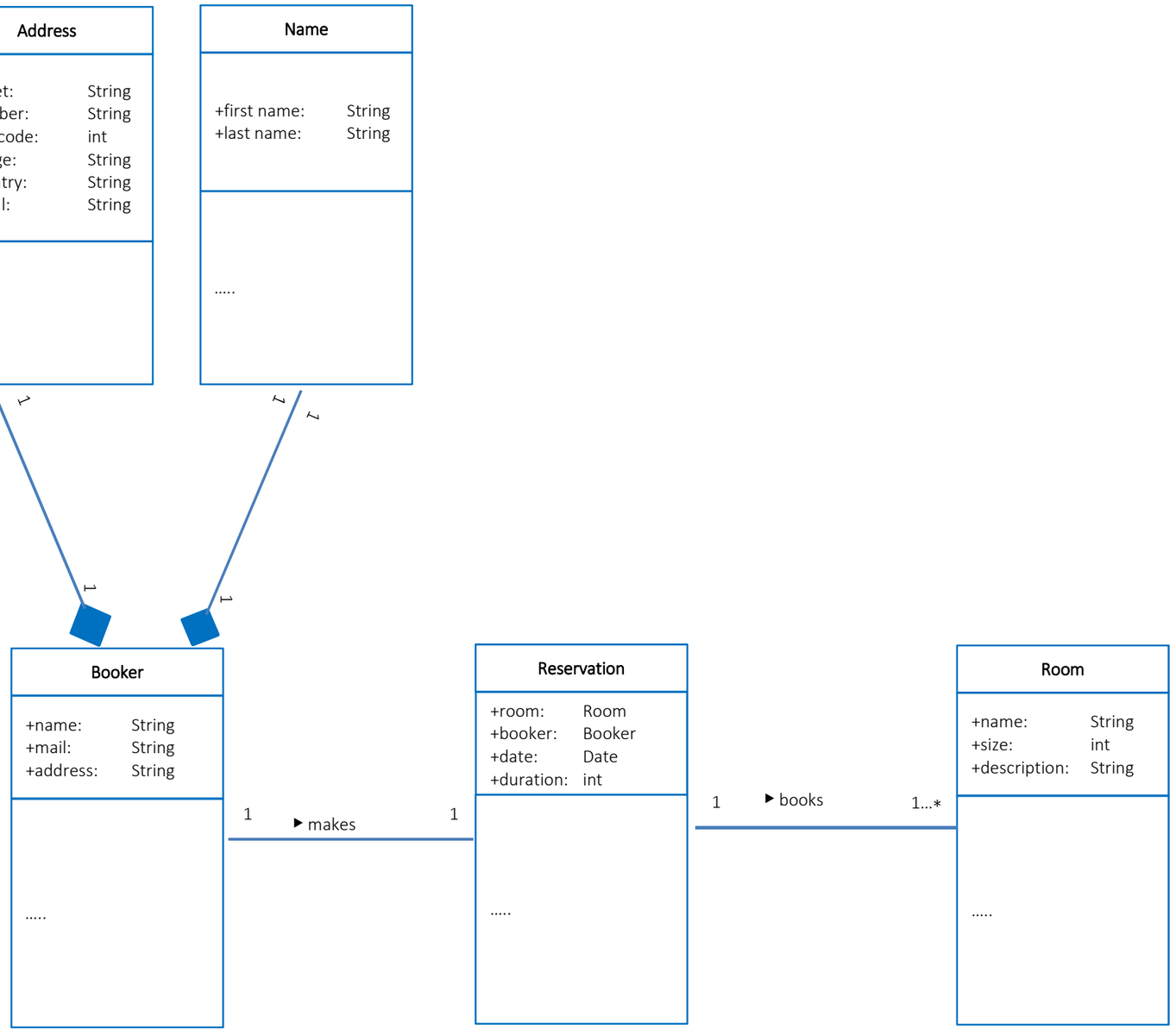
Template



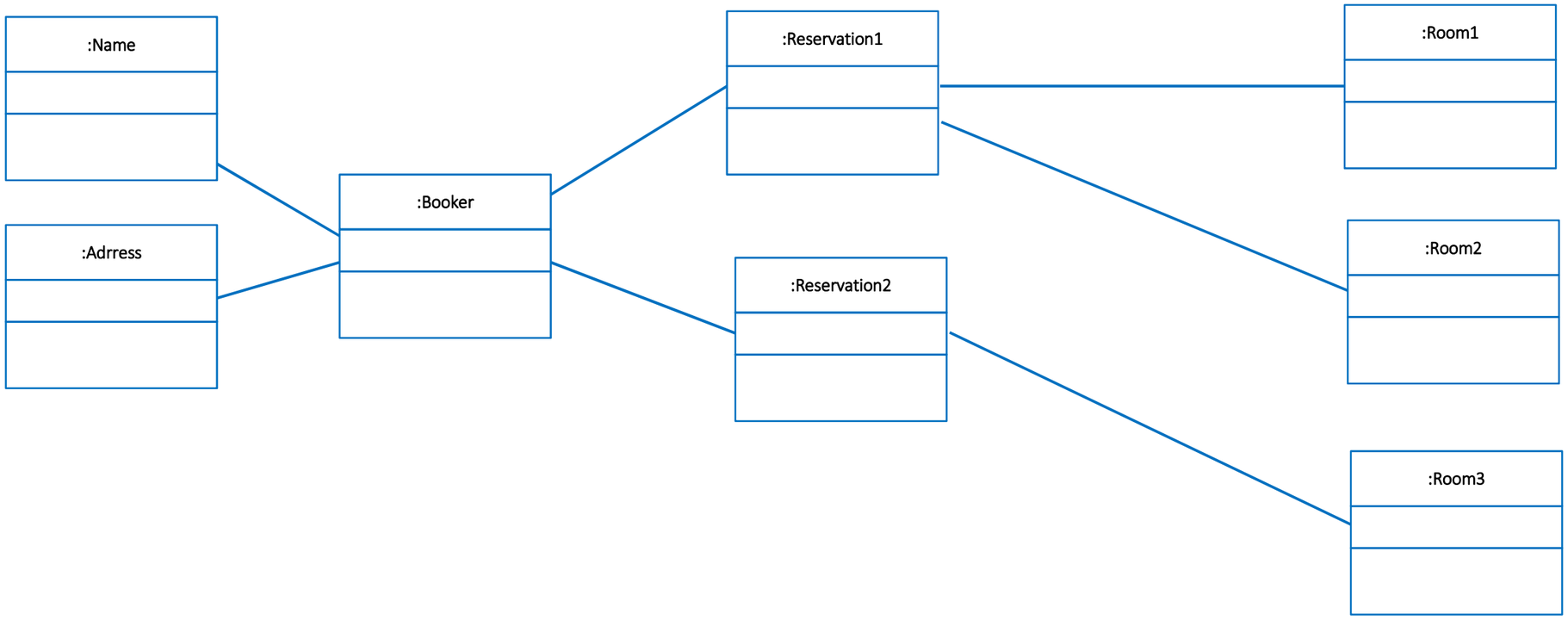


Domain Model

Class diagram

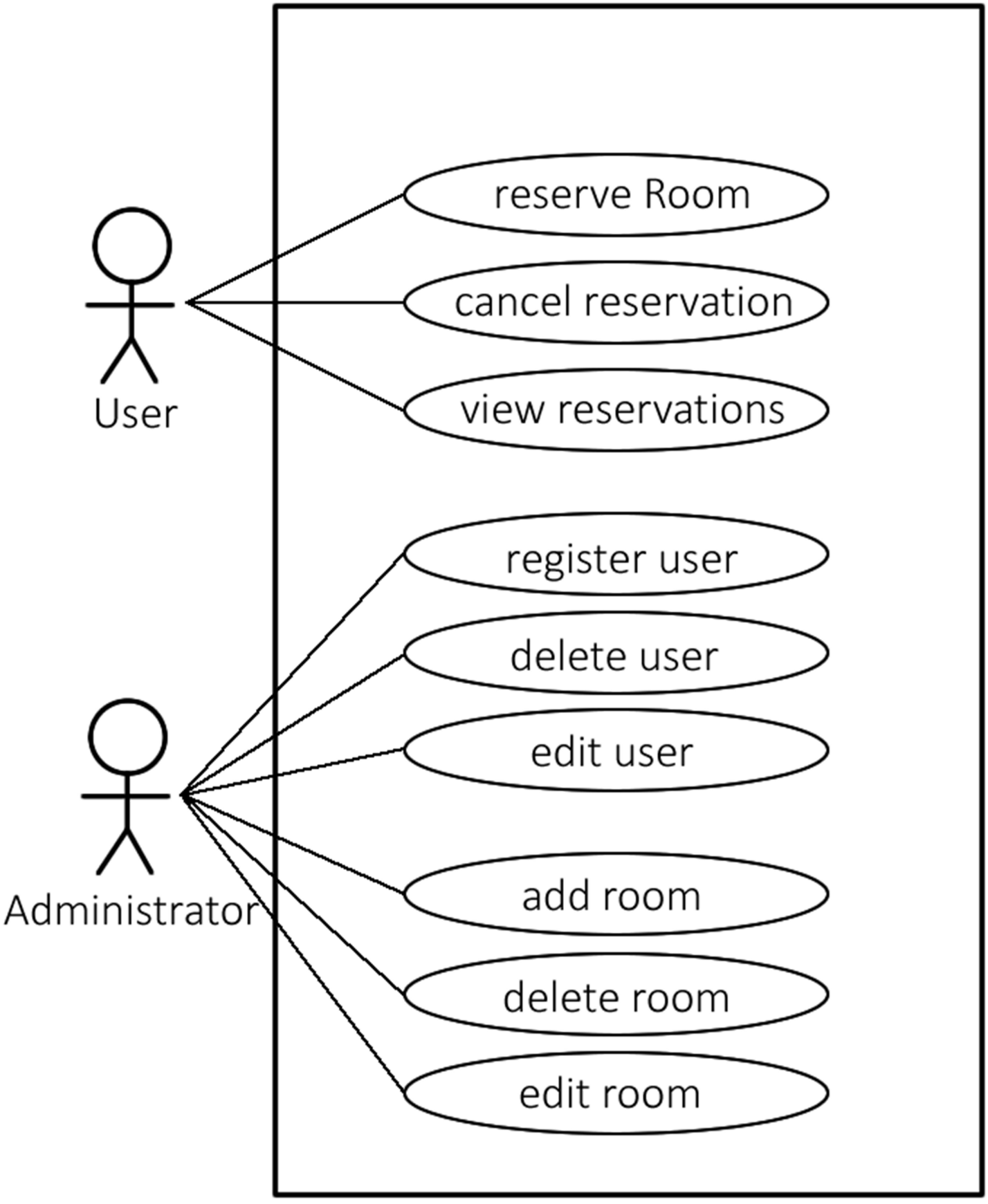


Object diagram



Context Diagram

Use Case



User Stories

Title	Description	Success	Fail
Register user	As an administrator, I want to register new users, so they can reserve rooms.	Users can only book rooms, if they are registered.	The reason of error will be shown.
Delete user	As an administrator, I want to delete a user, so he can't reserve rooms anymore.	Users, who don't have the permit to book rooms, must be deleted.	The reason of error will be shown.
Edit user	As an administrator, I want to edit a user, so the data are up to date.	Data from the users have to be up to date.	The reason of error will be shown.
Insert room	As an administrator, I want to insert a room, so users can reserve it.	Rooms can only be reserved if they are inserted in memory.	The reason of error will be shown.
Delete room	As an administrator, I want to delete a room, so users can't reserve it anymore.	Rooms which do not exist anymore have to be deleted, so they can't be booked by users.	The reason of error will be shown.
Edit room	As an administrator I want to edit a room, so users can either reserve again or don't reserve anymore.	Rooms are not always for booking, perhaps there is a renovation to do and then user cannot book these rooms.	The reason of error will be shown.
Reserve room	As a user, I want to reserve a room, so a booker can utilize it.	When bookers want to utilize a room, they have to be reserved in advance.	The reason of error will be shown.
Cancel room	As a user, I want to cancel a reservation, so it's available again.	When bookers have booked a room, they may be cancel the reservation and this have to be adapted in the reservation-memory.	The reason of error will be shown.
View room reservations	As a user, I want to see all reservations of a room, so I can see which rooms are occupied.	To see, which rooms are free for bookers, I need a view of all free / occupied rooms.	The reason of error will be shown.
View booker reservations	As a user, I want to see all reservations from a booker, so I can calculate which customers book often rooms.	To see, which bookers book often rooms, I need a view with the reservations form the past.	The reason of error will be shown.

Glossary

General

SRS	Smart Reservation System

Artefacts

Sprint Backlog	Items/stories from Product Backlog (only a few hours)
Release Backlog	A number of items from the Product Backlog is selected for each release
Product Backlog	Product Owner presents Product Backlog with all relevant user stories. Also the priority is mentioned.
User Stories	A written description of a functionality valuable to either a user (or owner) that will use the software system in future.
Use Case	Use cases are written stories.

Stakeholders

Development Team	The Development Team is responsible for developing functionality and for delivering shippable increments of the SRS at the end of each Sprint. Developer Teams are self-managing, self-organizing, and cross-functional and they are responsible for figuring out how to turn Product Backlog into an increment of functionality within iteration and managing their own work to do so.
Scrum Master	Person who builds the relationship between customer and development-team. Scrum Master participates on all Scrum-Meetings. The Scrum Master is responsible for the Scrum process, facilitate the Team and protect the team from outside noise.
Product Owner	The Product Owner represents the product's stakeholders and the voice of the customer and is accountable for ensuring that the team delivers value to the business. The Product Owner writes customer-centric items (user stories), prioritizes them based on importance and dependencies, and adds them to the Product Backlog. The Product Owner is responsible for maximizing the ROI. The Product Owner priories the Product Backlog
Customer	DUE1
Investor	DUE1

User groups

User	*Persons who will use the SRS and book rooms for the booker.*
Administrator	*Persons who has the responsibilities of all users and all rooms (register new rooms, edit rooms, delete rooms)*
Booker	*The person who will book a room.*