Project iBeacon

ProjDat 2015

Report 1

March 23, 2015

Emil Slot Arakelian Jensen - 170494

Samuel Korn - 101186

Mark Jay Nielsen - 250694

Rune Bjerg Ono - 180187

TA: Kasper Passov

Contents

1 Problem definition

How many people are in the office?
Use iBeacons to determine the amount of people currently in the office. Aggregate the data to apply machine learning and data visualizations. Using iBeacons we would like to make an app that registers whenever the user enters or leaves our office. We would like to aggregate that data to use for cool data visualizations and applying machine learning to the data to obtain knowledge about the general usage patterns of the office. This could e.g. be used to adjust the heating, automatically lock the doors and turn off the lights when the last person leaves, predict when we need to order more or less lunch, or simply to check whether a certain person is currently at the office.[1]

The above text, is the original description of one of the many projects made available to us from the company Shape A/S.

On March the 12th, we had a meeting with Søren Ulrikkeholm, who is the developer responsible for student contact at Shape. We talked about the different projects and realistic acceptance criteria, taking into account our limited experience. We settled on the aforementioned project, because it was deemed the most appropriate for our level of skill, as well as being the one in which Shape had the most interest.

Søren explained that the company's biggest interest is learning how to use iBeacons with Android, with aggregation of the data and application of machine learning and data visualization are secondary objectives. We have decided to split the project into two development phases.

Our primary focus is on the first development phase, which consists of developing an app for Android that registers individual users entering and leaving the office space.

The second development phase consists of developing the backend for the system and integrating it with the app, in order to aggregate the data from users entering and leaving the office in a database.
A third phase, which we will not be working on, would be to apply machine learning and data visualizations to obtain knowledge about general usage patterns in the office.

2 Initial Software Project Management Plan and Initial software architecture

As of now, we are only in the initial phase of designing our app, therefore the current management plans and schedules are rough drafts.

In terms of responsibilities, we do not plan to appoint a project manager. We are a group of only four people and are confident that we do not need to appoint someone to supervise the remaining three. We will however, later on in the development, create detailed schedules and individual deadlines for smaller assignments, which we will use to keep track of our progress rather than having a single person be manager. This approach is not quite the same as mentioned in our textbook,[2] however, given the size of our team we do not see how setting up a textbook SCRUM team will have any major benefits.

We do not plan to have any expenses tied to the development of our project, nor does the development software pose any costs. Shape A/S will provide us with access to the iBeacon devices.

As of this moment we do not know exactly how we will design the system. We have not yet seen the iBeacons, nor do we know exactly how they work. It is therefore hard to know how to build a system around them. We have made a very rough draft of a partial work breakdown structure for the project, but it is still very general and not very detailed.

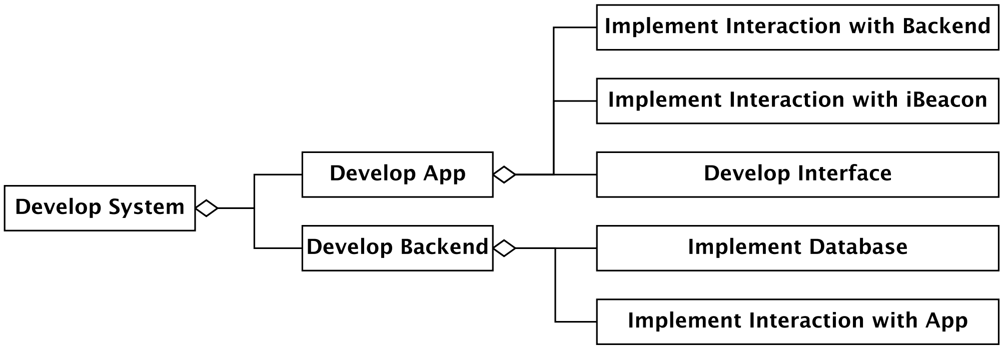
Develop System
Develop App
Develop Backend
Implement Interaction with Backend
Implement Interaction with iBeacon
Develop Interface
Implement Database
Implement Interaction with App

Fig 1: Partial work breakdown structure for the project.

3 Project Agreement definition

The following is a translation of the Project Agreement definition, agreed upon (in its original danish language version) by both Søren Ulrikkeholm of Shape A/S and ourselves:

Together with Shape A/S we have chosen to work on project 5 from their website www.shape.dk/projects. The project consists of developing an app for the Android and iOS operating systems that, using an iBeacon device, collects data showing how many people are present in their office at a given time. This data is collected with the intention of finding usage patterns which could potentially be used for automatic control of a number of functions in the office (e.g. turning the lights on and off, regulating the room temperature etc.).

During our meeting with Søren Ulrikkeholm of Shape A/S on March 12th we discussed how the project can be split into 3 development phases. First phase involves designing and developing an app which collects data fra an iBeacon. Second phase is to integrate the app with a backend database that stores all the collected data, and the third phase revolves around the use of the stored data for data visualizations and machine learning purposes.

An app for both iOS and Android would be too much for us to handle, which is why we have agreed to only develop the app for Android. Our limited programming experience means that we will be focusing solely on the first two development phases, i.e. we will only be designing and developing the app which gathers data from iBeacons, and implementing the database to store all the data. Additional functionality might be developed if we have time.

An outline of our schedule for the project follows:

March 16 - March 22: Problem definition
March 23 - April 5: Initial design phase
April 5 - June 7: Main development phase
June 8: Product delivery date

We have made an arrangement with Shape A/S to be able to work on the project in their office one day of every week.

Our next meeting with Søren Ulrikkeholm will be on March 25th at 10 o'clock, where we will have our first look at the iBeacon device.

Aside from their time Shape A/S will have no expenses tied to this project.

4 Exercises

4.1 1-6.

Specify which of these statements are functional requirements and which are nonfunctional requirements
Functional requirements:
- "The TicketDistributor must enable a traveler to buy weekly passes."
- "The TicketDistributor must always be available."
- "The TicketDistributor must provide a phone number to call when it fails."

Nonfunctional requirements:
- "The TicketDistributor must be written in Java."
- "The TicketDistributor must be easy to use."

4.2 1-8.

In the following description, explain when the term account is used as an application domain concept and when as a solution domain concept:
"...managing bank accounts for mobile costumers." - It is used as an application domain concept, saying what it is supposed to do. "...provide access to the accounts when the..." - It is used the same way as before. "One proposal is that accounts are made available on the mobile computer..." - Here it is used as a solution domain concept, as it is a possible solution to be evaluated. "...the accounts show the amounts from the last connected session." - It is used in connection with the previous use.

4.3 2-6, 2-7, 2-9.

Draw a class diagram representing a book. Add multiplicity to the class diagram you produced in Exercise 2-6. Extend the class diagram to include attributes.

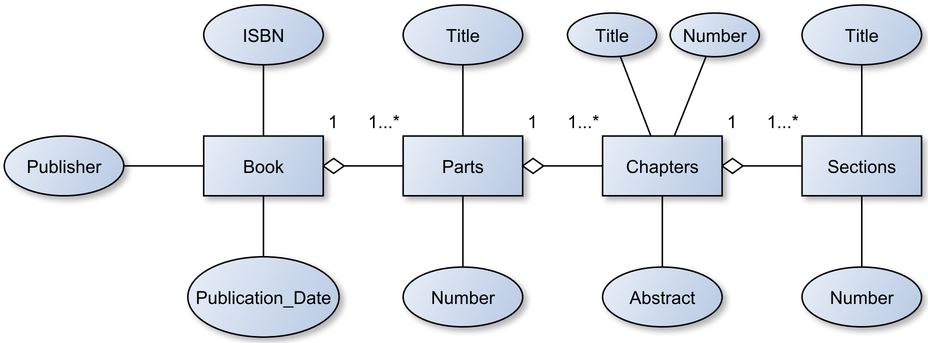

Fig 2: Class diagram for exercise 2-6, 2-7 and 2-9

4.4 2-10.

Add an abstract class and an inheritance relationship to factor out these two attributes into the abstract class

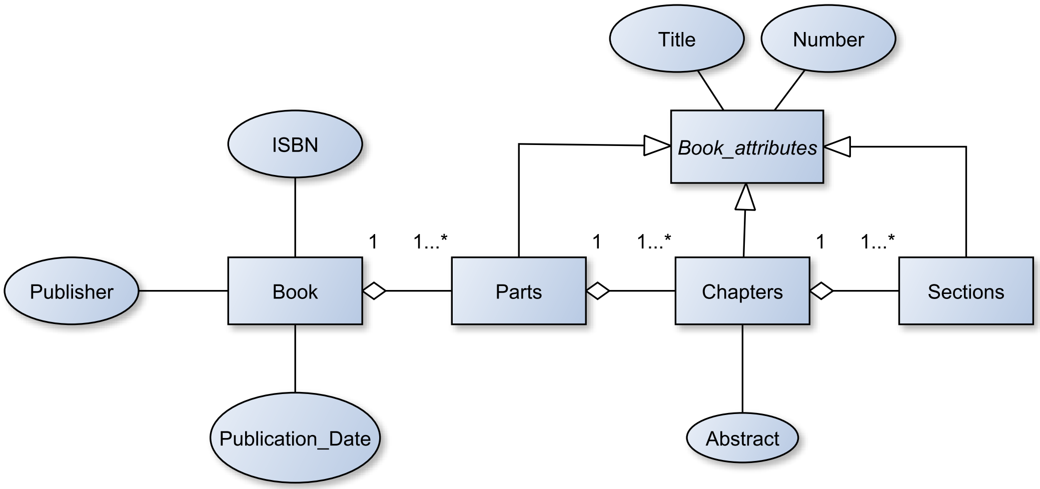

Fig 3: Altered class diagram for exercise 2-10

4.5 5-3.

Arrange the objects listed in Exercises 5-1 and 5-2 horizontally on a sequence diagram.

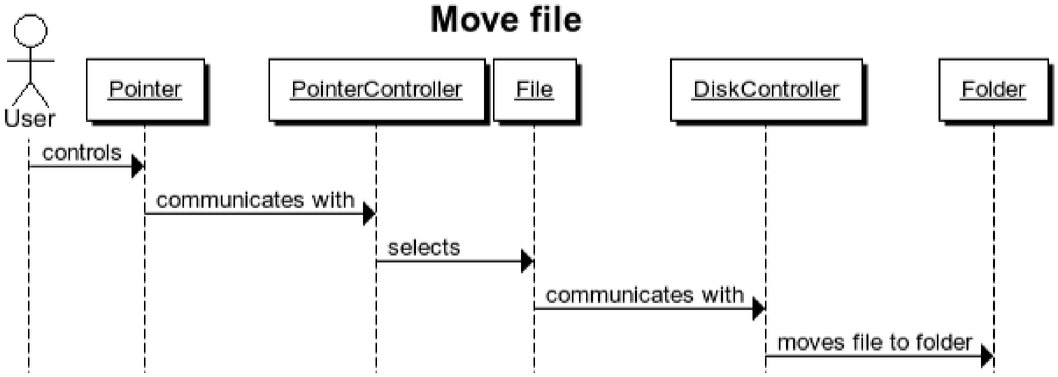

Fig 4: Sequence diagram for exercise 5-3

4.6 7-1

Draw a UML deployment diagram representing the hardware/software mapping.

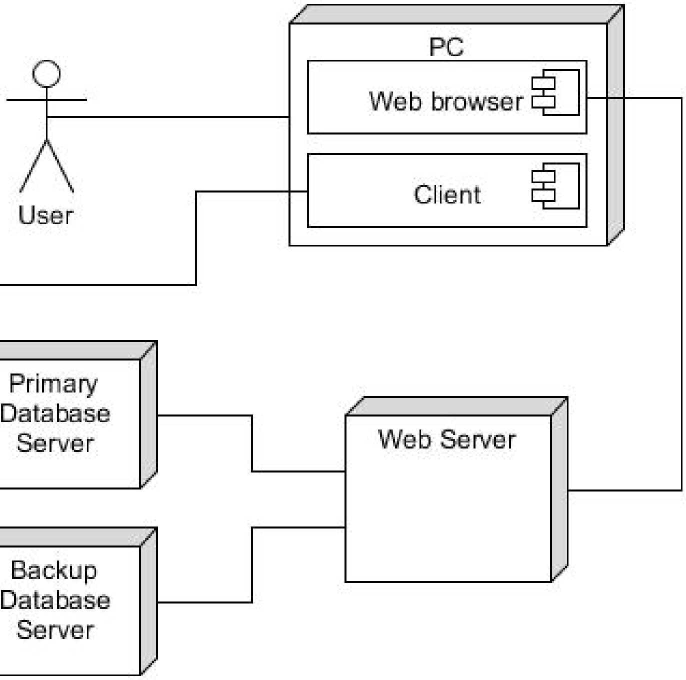

Fig 5: Deployment diagram for exercise 7-1

References

[1] www.shape.dk/projects

[2] Bruegge & Dutoit (2010), Object-Oriented Software Engineering - using UML, Patterns and Java, Third edition, Pearson 2010. Page 611