

Sensur av hovedoppgaver

Høgskolen i Sørøst-Norge

Fakultet for teknologi og maritime fag



Prosjektnummer: **2017-15**

For studieåret: **2016/2017**

Emnekode: **SFH03201**

Prosjektnavn

Beacon

Utført i samarbeid med: Flatirons Solutions

Ekstern veileder: Roger Werner Laug, Eirik Nordstrand, Eirik André Eidså

Sammendrag: In the field of technical documentation there is a need for reducing human errors and increasing efficiency to ensure economic growth.

In this project, we explore the possibility of using Bluetooth low energy beacons to filter applicable content in a technical manual.

This project presents a solution showing how to filter technical documentation using Bluetooth low energy beacons, and contributes with research on how to solve these challenges.

The team have created a component-based application which enhances the process of applying a contextual filter using Bluetooth low energy beacons.

Stikkord:

- Bluetooth low energy beacons
- Tekniske dokumentasjonssystemer
- Mobilapplikasjon

Tilgjengelig: JA

Prosjekt deltagere og karakter:

| Navn | Karakter |
|-----------------------|----------|
| Afshin Alavi | |
| Daniel Idris | |
| Ole Christian Moholth | |
| Tor Egil Øverberg | |

Dato: 8. Juni 2017

Radmila Juric
Intern Veileder

Olaf Hallan Graven
Intern Sensor

Roger Werner Laug
Ekstern Sensor

A small step in coding, a giant leap in engineering

BEACON

Group 15

Kongsberg, 24. May 2017

How beacons can be used to simplify the process of searching through a publication from CORENA Pinpoint.

THE USE OF
BLUETOOTH LOW
ENERGY BEACON TO
FILTER
DOCUMENTATION

FLATIRONS®

HSN Høgskolen
i Sørøst-Norge

ABSTRACT

In the field of technical documentation there is a need for reducing human errors and increasing efficiency to ensure economic growth.

In this project, we explore the possibility to use Bluetooth low energy beacons to filter applicable content in a technical manual.

This project presents a solution showing how to filter technical documentation using Bluetooth low energy beacons, and contributes with research on how to solve these challenges.

The team have created a component-based application which enhances the process of applying a contextual filter using Bluetooth low energy beacons.

This is a contribution to economy, safety and efficiency when using technical documentation, where such use is crucial.

TABLE OF CONTENTS

| | | |
|----------|------------------------------------------------|-----------|
| 1 | Introduction | 11 |
| 1.1 | Background..... | 11 |
| 1.2 | Group Members..... | 11 |
| 1.3 | Document structure | 13 |
| 2 | The Problem Domain..... | 15 |
| 2.1 | Flatirons Solutions | 15 |
| 2.2 | Scenario..... | 16 |
| 2.3 | CORENA Pinpoint | 16 |
| 2.4 | Our Assignment | 17 |
| 3 | Theoretical Background | 19 |
| 3.1 | About Angular..... | 19 |
| 3.1.1 | Routing in Angular | 19 |
| 3.1.2 | Application..... | 19 |
| 3.2 | About Nativescript | 20 |
| 3.2.1 | What is a NativeScript plugin..... | 20 |
| 3.3 | About Typescript..... | 20 |
| 3.4 | Bluetooth Low Energy | 21 |
| 3.4.1 | BLE Beacons | 21 |
| 3.4.2 | RSSI (Received Signal Strength Indicator)..... | 21 |
| 3.5 | Estimote | 22 |
| 3.6 | S1000D Specification..... | 22 |
| 3.6.1 | Applicability | 22 |
| 3.7 | CORENA Pinpoint Neutral package..... | 23 |
| 3.8 | Test Driven Development..... | 24 |
| 3.9 | Unit Testing | 25 |
| 3.9.1 | JUnit | 25 |
| 3.10 | Agile Modeling and Development: Scrum..... | 25 |
| 3.10.1 | Daily Scrum Meeting | 26 |
| 3.10.2 | User Stories | 26 |
| 3.10.3 | Product Backlog..... | 26 |
| 3.10.4 | Sprint | 26 |
| 3.11 | Risk..... | 29 |
| 3.12 | Prototyping..... | 29 |
| 4 | Methodology | 31 |
| 4.1 | How Scrum has been used | 31 |
| 4.2 | Group dynamics | 31 |
| 4.3 | Communication with Flatirons Solutions | 32 |
| 4.4 | Meetings with Internal Supervisor | 32 |
| 4.5 | Seating Arrangement | 32 |
| 4.6 | Risk..... | 32 |
| 4.6.1 | Identifying Risks..... | 32 |

| | | |
|-------------|----------------------------------------------------|-----------|
| 4.6.2 | Developing Risk Response Strategies | 33 |
| 4.6.3 | Monitoring and Controlling Risks | 33 |
| 4.6.4 | Risk Matrix | 33 |
| 4.7 | Testing | 36 |
| 4.8 | NativeScript Application Architecture | 37 |
| 4.8.1 | Component | 37 |
| 4.8.2 | Template | 37 |
| 4.8.3 | Service | 37 |
| 4.9 | Style Guide | 38 |
| 4.10 | JSDoc | 38 |
| 5 | Requirements | 39 |
| 5.1 | Functional | 40 |
| 5.1.1 | Swipe | 40 |
| 5.1.2 | detect beacons | 40 |
| 5.1.3 | Filter HTML components | 41 |
| 5.1.4 | Beacon list ui | 41 |
| 5.2 | Non-Functional | 42 |
| 5.2.1 | DC9 data set | 42 |
| 5.2.2 | Cross platform | 42 |
| 6 | Modeling the Product | 43 |
| 6.1 | Use Case | 44 |
| 6.2 | Sequence Diagram | 45 |
| 6.2.1 | Choose Publication | 45 |
| 6.2.2 | Choose Document | 45 |
| 6.2.3 | Swipe | 46 |
| 6.2.4 | Select Beacon | 46 |
| 6.3 | Software Architecture | 47 |
| 7 | The Implementation | 49 |
| 7.1 | Publication Manager | 49 |
| 7.1.1 | Creating a Publication | 50 |
| 7.1.2 | Loading a publication | 50 |
| 7.1.3 | Documents | 52 |
| 7.1.4 | Table of Contents | 53 |
| 7.2 | Filter Component | 53 |
| 7.2.1 | S1000D Evaluating filter statements | 54 |
| 7.2.2 | Filter Component Class Diagram | 56 |
| 7.2.3 | Evaluating Filter Statement | 57 |
| 7.2.4 | Applying Filter Statement | 57 |
| 7.3 | Document Viewer | 58 |
| 7.3.1 | Loading Document From Publication | 58 |
| 7.3.2 | Displaying Document In WebView | 59 |
| 7.3.3 | WebView | 59 |
| 7.3.4 | Pinpoint Document View | 60 |

| | | |
|------------|--------------------------------------------|-----------|
| 7.3.5 | HTML document factory..... | 63 |
| 7.4 | Toc View | 65 |
| 7.4.1 | Loading Table Of Contents | 65 |
| 7.4.2 | Toc Nodes | 65 |
| 7.4.3 | Navigating the table of contents | 66 |
| 7.5 | Beacon Component..... | 67 |
| 7.5.1 | Estimote Plugin | 68 |
| 7.5.2 | Beacon detection for iOS..... | 70 |
| 7.5.3 | Beacon Detection for android | 70 |
| 7.5.4 | Preparing beacon Data | 71 |
| 7.5.5 | Beacon View | 73 |
| 7.6 | Folder Structure | 74 |
| 8 | Evaluation | 75 |
| 8.1 | Have we met the Requirements? | 75 |
| 8.2 | Challenges | 75 |
| 8.2.1 | Integrating beacons in NativeScript..... | 75 |
| 8.2.2 | Implementing Swipe on android | 76 |
| 8.2.3 | New development tools | 76 |
| 8.2.4 | No one has done it before | 76 |
| 8.2.5 | DC9 challenge with iOS..... | 76 |
| 8.3 | Practical Evaluation..... | 76 |
| 8.3.1 | Estimote Beacons | 76 |
| 8.3.2 | Usage of Nativescript..... | 77 |
| 8.3.3 | Beacon Accuracy..... | 77 |
| 8.4 | Demonstration..... | 80 |
| 9 | Conclusion | 83 |
| 9.1 | Contribution | 84 |
| 9.1.1 | Functional for Flatirons Solution | 84 |
| 9.1.2 | Precedence | 84 |
| 9.1.3 | Knowledge | 84 |
| 9.1.4 | Future Work..... | 84 |
| 9.1.5 | Novelty | 84 |
| 9.1.6 | Documentation Management | 84 |
| 9.1.7 | Safety and Economy | 84 |
| 9.2 | Related Work..... | 85 |
| 9.3 | Future Work..... | 85 |
| 9.3.1 | Beacon Location | 85 |
| 9.3.2 | Implementation | 91 |
| 9.3.3 | Component Identification..... | 93 |
| 9.3.4 | Storing Filter ID on Beacons | 93 |
| 9.4 | Considerations | 94 |
| 9.4.1 | Power Consumption | 94 |
| 9.4.2 | Multi Threading | 94 |

| | | |
|----------------------------------|--------------------------------------|-----------|
| 9.4.3 | Security | 94 |
| 9.5 | Other possible solutions..... | 94 |
| 9.5.1 | WiFi..... | 94 |
| 9.5.2 | Augmented Reality | 95 |
| 9.5.3 | Machine Learning | 95 |
| 10 | References | 97 |
| Appendix A – User Manual | | |
| Appendix B – Decisions | | |
| | NativeScript Decision | |
| | Beacon Decision | |
| | Estimote plugin decision | |
| Appendix C – Sprint | | |
| Appendix D – Tests | | |
| | Publication Manager | |
| | Applicability | |
| | Beacon | |
| Appendix E – Budget | | |
| Appendix F – TimeSheets | | |
| Appendix G – Timeline | | |
| Appendix H – User Stories | | |

TABLE OF FIGURES

| | |
|----------------------------------------------------------------------|----|
| Figure 1 Technician [1] | 16 |
| Figure 2 Corena suite supports S1000D [2] | 17 |
| Figure 3 Angular logo [4] | 19 |
| Figure 4 NativeScript logo [8] | 20 |
| Figure 5 TypeScript [9] | 20 |
| Figure 6 Bluetooth Logo [78]..... | 21 |
| Figure 7 - Beacons [11] | 21 |
| Figure 8 Estimote [13] | 22 |
| Figure 9 - Folder structure of CORENA Pinpoint Neutral Package | 23 |
| Figure 10 - TDD process [18] | 24 |
| Figure 11 - Scrum Framework [22] | 25 |
| Figure 12 – Burndown chart from Sprint 8..... | 28 |
| Figure 13 - Risk Matrix, based on risk matrix from [25] | 33 |
| Figure 14 - Angular architecture [25] | 37 |
| Figure 15 DC9 Airplane [27] | 39 |
| Figure 16 - Use Case model | 44 |
| Figure 17 Publication class diagram | 49 |
| Figure 18 - Creating a documentFolder..... | 52 |
| Figure 19 - Sequence diagram loading table of contents..... | 53 |
| Figure 20 - Class diagram for evaluating filter statements..... | 56 |
| Figure 21 jQuery for filtering documentation | 62 |
| Figure 22 - Class diagram for beacon view | 67 |
| Figure 23 Estimote SDK [28] | 68 |
| Figure 24 - iOS class diagram for ranging beacons..... | 70 |
| Figure 25 - Getting filter statement | 72 |

| | |
|------------------------------------------------------------------------|-----|
| Figure 26 Display beacon list..... | 73 |
| Figure 26 - Folder Structure | 74 |
| Figure 28 - unfiltered signal..... | 78 |
| Figure 29 – 10 th order low pass filter | 79 |
| Figure 30 – 20 th order low pass filter | 79 |
| Figure 31 - Measuring method [25] | 87 |
| Figure 32 - Triangulation with multiple intersection points [25] | 87 |
| Figure 33 - Centroid positioning concept [25]..... | 89 |
| Figure 33 - Toc view of bike data set..... | 107 |
| Figure 35 - Beacon view | 108 |
| Figure 35 - Document view | 109 |
| Figure 36 - Document view (Filter Applied)..... | 110 |
| Figure 37 - Estimote Beacons..... | 112 |
| Figure 38 - Sprint 1 issues..... | 115 |
| Figure 39 - Sprint 1 burndown | 115 |
| Figure 40 - Sprint 1 retrospective..... | 116 |
| Figure 41 - Sprint 2 issues..... | 117 |
| Figure 42 - Sprint 2 burndown | 117 |
| Figure 43 - Sprint 2 Retrospective | 118 |
| Figure 44 - Sprint 3 Issues | 119 |
| Figure 45 - Sprint 3 Burndown | 119 |
| Figure 46 - Sprint 3 Retrospective | 120 |
| Figure 47 - Sprint 4 Issues | 121 |
| Figure 48 - Sprint 4 Burndown | 121 |
| Figure 49 - Sprint 4 Retrospective | 122 |
| Figure 50 - Sprint 5 Issues | 123 |

| | |
|-------------------------------------------|-----|
| Figure 51 - Sprint 5 Burndown | 123 |
| Figure 52 - Sprint 5 Retrospective | 124 |
| Figure 53 - Sprint 6 Issues | 125 |
| Figure 54 - Sprint 6 Burndown | 125 |
| Figure 55 - Sprint 6 Retrospective | 126 |
| Figure 56 - Sprint 7 Issues | 127 |
| Figure 57 - Sprint 7 Burndown | 127 |
| Figure 58 - Sprint 7 Retrospective | 128 |
| Figure 59 - Sprint 8 Issues | 129 |
| Figure 60 - Sprint 8 Burndown | 129 |
| Figure 61 - Sprint 8 Retrospective | 130 |
| Figure 62 - Sprint 9 Issues | 131 |
| Figure 63 - Sprint 9 Burndown | 131 |
| Figure 64 - Sprint 9 Retrospective | 132 |
| Figure 65 - Sprint 10 Retrospective | 132 |
| Figure 66 - Sprint 10 Burndown | 132 |
| Figure 67 - Sprint 10 Retrospective | 133 |
| Figure 68 - Sprint 11 Issues | 134 |
| Figure 69 - Sprint 11 Burndown | 134 |
| Figure 70 - Sprint 11 Retrospective | 135 |
| Figure 71 - Sprint 12 Issues | 136 |
| Figure 72 - Sprint 12 Burndown | 136 |
| Figure 73 - Sprint 12 Retrospective | 137 |
| Figure 74 - Sprint 13 Issues | 138 |
| Figure 75 - Sprint 13 Burndown | 138 |
| Figure 76 - Sprint 13 Retrospective | 139 |

1 INTRODUCTION

This bachelor assignment investigates enhancing the user experience in document viewing with the use of beacons.

There is a need for a safer, faster, more intuitive and cost saving document user experience in areas where safety and economy are crucial factors. One example is maintenance in the airline industry.

When the team first started to look for possible projects for our bachelor project we did not have anything special in mind. Tor Egil contacted Flatirons Solutions, and they proposed a project involving Bluetooth low energy beacons.

On this basis, Flatirons Solutions suggested the idea of using beacons to filter content in a data set. The purpose of this is to minimize errors at aircraft maintenance, use less man hours and cut costs. A cross platform mobile application was developed by the bachelor group. The application is called BApp short for Beacon Application

1.1 BACKGROUND

We are four students studying computer science at the University College of Southeast Norway. Three of us have a background in embedded systems, while one has a background from virtual systems at the University College of Southeast Norway.

1.2 GROUP MEMBERS

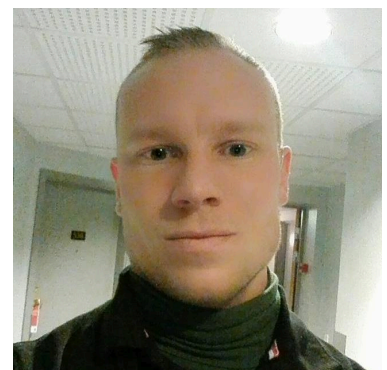
TOR EGIL ØVERBERG: PROJECT LEAD AND SCRUM MASTER

PROJECT LEAD

- Responsible for the management of the project.
- Ensures that project is on track

SCRUM MASTER

- Responsible for making the team self-organized
- Remove any impediments for the development team
- Ensures we follow the scrum framework
- Helping the product owner effectively manage the product backlog



OLE CHRISTIAN MOHOLTH: TECH LEAD AND TEST LEAD

TECHNICAL LEAD

- Responsible for the architecture for the software application
- Ensures we follow correct standards for coding
- Act as a mentor for the other developers

TEST LEAD

- Responsible for ensuring proper testing
- Ensures we follow the Test-Driven Development Guidelines



DANIEL IDRIS: PRODUCT OWNER

- Responsible for the product backlog
- Ensures the product backlog is clearly defined
- Organizing the product backlog for to best achieve goals
- Ensures we deliver the correct product



AFSHIN ALAVI: DOCUMENTATION

- Responsible for all our documentation work
- Ensures files are up to date and structured
- Ensures that the development team documents their solutions properly
- Ensures the internal wiki page is up to date



1.3 DOCUMENT STRUCTURE

CHAPTER 2 – THE PROBLEM DOMAIN

This chapter describes the domain in which the research took place. It presents the basis for the assignment, and the research question for which we present an answer.

CHAPTER 3 – THEORETICAL BACKGROUND

Here we will introduce the theory behind our project. You will get a quick introduction to

- The development frameworks used during the project
- The scrum model
- The S1000D specification
- Bluetooth

CHAPTER 4 – METHODOLOGY

Here we explain how we have adapted the tools to our project

CHAPTER 5 – REQUIREMENTS

Requirements given to us by flatirons, limitations

Here we introduce the prototype we have developed. You'll get an introduction to the architecture and models we have developed for this prototype.

CHAPTER 6 – MODELING THE PRODUCT

Follows the process of developing the system architecture from the use case diagrams

CHAPTER 7 – THE IMPLEMENTATION

Details of how the architecture was implemented

CHAPTER 8 – EVALUATION

Here we discuss the result of the project and experience gained

CHAPTER 9 – CONCLUSION

This chapter talks about related work, our contributions, considerations when developing the product and future work required.

CHAPTER 10 – REFERENCES

This chapter contains the list of all references used throughout the documents. This includes image sources.

APPENDIX A – USER MANUAL

This is a manual for using the application.

Topics include how to install and how to use the application

APPENDIX B – USER STORIES

Here we list the user stories during the project

APPENDIX C – SPRINTS

Here we present the sprint reports and what we accomplished every sprint

APPENDIX D – TESTS

Here we present the unit tests, and more detailed information of how we tested the functionalities

APPENDIX E – BUDGET

Here we present the budget

2 THE PROBLEM DOMAIN

This chapter describes the domain in which the research took place. It presents the basis for the assignment, and the research question for which we present an answer.

2.1 FLATIRONS SOLUTIONS

Flatirons Solutions provide consulting, software and outsourcing of solutions to content intensive industries using their proven approach to Content Lifecycle Management(CLM). For more than 20 years they have served global fortune 1000 companies in aerospace, automotive, electronics, financial services, government, healthcare and publishing. They have their headquarters located in Irvine, California with offices around the world in US, China, the UK, Scandinavia, Germany and India. Worldwide they are a team of 450 professionals. Flatirons Solutions was formerly known as Infotrust Group, Inc. In 2013, they changed their name to Flatirons Solutions, Inc.

- The Bachelor Student team working on the Beacon Project is connected to Flatirons Solutions Norway.
- In this office, most of the activities are focused R&D (Research and Development)
 - Representatives from Sales, Customer Support, IT and Finance are also located in the office.
- Flatirons Solutions R&D department is around 70-75 people
 - Managed from US, both in Headquarter Irvine – California and in Boulder – Colorado.
 - Other R&D Offices are in Shanghai – China, Chennai – India, Kongsberg – Norway
- R&D is organized in Development Teams
 - 4 in Shanghai
 - 2 in Norway
 - 1 in India
- Products are delivered to a specific Solution, Fleet or OEM.
 - Fleet solution is for Operators, for example an Airline like British Airways
 - OEM solution is for Manufacturers, for example an Aircraft OEM like Airbus or Boeing

2.2 SCENARIO

We are making software application for accessing documentation of nearby physical components on a mobile device. A possible scenario is an airplane technician who is going to perform maintenance on an airplane. He accesses all the required documentation on his mobile phone when he is near the part of the plane he is scheduled to maintain. The software application works with special markers on the plane called beacons which give the mobile phone its identity number. The mobile phone will search through a data set to get the information on that specific identification number given by the beacon. This will give the technician access to all information such as history, measurements, parts and how to maintain them.



Figure 1 Technician [1]

2.3 CORENA PINPOINT

CORENA Pinpoint is generic documentation manual viewer developed by Flatirons Solutions. The document viewer helps companies search through all their documentation. Flatirons Solutions have developed a rich search function that lets you search for specific words or applicability. They have a mobile and a desktop application for their software.

“Various time and movement studies conducted by Defense Logistics Organizations (DLOs) and Fleet Operators have found that up to 40% of engineer and mechanic work time is used looking for the technical information required to perform tasks, or documenting details of completed tasks after completion. CORENA Pinpoint Mobile cuts this time at least in half, realizing millions in productivity and capital savings for Fleet Operators.” “including both S1000D and iSpec 2200 doc types.” [2]



Figure 2 Corena suite supports S1000D [2]

2.4 OUR ASSIGNMENT

Can Bluetooth low energy beacons be used to filter content in a mobile app for technical publications?

The assignment is how to filter content in a web viewer with the use of beacons. To do this we will make a mobile application with a preloaded generic data set. The application will run on both android and iOS.

The waypoints to achieve this goal will be:

- Loading of data set
- Filter content
- Connect to beacons
- Filter content with beacon data

The purpose of the assignment is to make documentation more accurate and applicable. This is beneficial for several technical industries where accuracy and accessibility to documentation will cut down on man hours and increase safety.

3 THEORETICAL BACKGROUND

Here we will introduce the theory behind our project. You will get a quick introduction to

- The development frameworks used during the project
- The scrum model
- The S1000D specification
- Bluetooth

3.1 ABOUT ANGULAR

Angular is a development platform for building mobile and desktop web applications using Typescript/JavaScript (JS) and other languages. Angular uses parts packaged together on what they do, such as if there are used for UI, service, modelling, communication etc. These packages are known as components. [3] [4]



Figure 3 Angular logo [4]

3.1.1 ROUTING IN ANGULAR

Routing is a function in Angular 2 that enables navigation between views within an application. Transferring to another view happens without creating a new page. The idea is borrowed from the browser where we use an URL address field and forward and backward buttons to navigate into different pages. [5]

3.1.2 APPLICATION

With the Angular Router object, we can pass routes as arguments to navigate between views. The routes can be bound to links or other input elements such as buttons and drop-down menus. Through the angular routing, it becomes easier to develop modular mobile applications.

3.2 ABOUT NATIVESCRIPT

NativeScript is a framework for creating native cross-platform applications for Android and iOS, with Universal Windows Platform support. [6] NativeScript apps are written using JavaScript or Typescript, and directly supports Angular 2. [7]

NativeScript uses native APIs for running code, and allows direct use of the native APIs.



Figure 4 NativeScript logo [8]

3.2.1 WHAT IS A NATIVESCRIPT PLUGIN

A NativeScript plugin is any npm package, published or not, that exposes a native API via JavaScript and consists of the elements as shown in

The main building blocks is the package.JSON and platforms folders. The package.JSON file specifies the name, version, description, dependencies etc. The platforms folders specify how to import the Estimote SDK for both iOS and android.

3.3 ABOUT TYPESCRIPT



Figure 5 TypeScript [9]

TypeScript is a typed superset of JavaScript that compiles to plain JavaScript. It is compatible with any browser, any host, any OS and it is open source. [9] Typings in Typescript lets developers use productive tools such as static checking and code refactoring when developing JavaScript applications.

3.4 BLUETOOTH LOW ENERGY

Bluetooth wireless technology is a short-range communications system intended to replace the cable(s) connecting portable and/or fixed electronic devices. [10]



Figure 6 Bluetooth Logo [78]

The design goal of BLE was to make a device with the lowest possible energy consumption, complexity and cost. While standard Bluetooth excels at handling lots of data at the cost of extra energy consumption. BLE excels at handling small amounts of data while still consuming minimal energy.

3.4.1 BLE BEACONS



Figure 7 - Beacons [11]

A BLE beacon is a small device that can transmit only a simple packet to more complex beacons able to transmit multiple packets at once even multiple different protocols with a range of up to 200 meters.

3.4.2 RSSI (RECEIVED SIGNAL STRENGTH INDICATOR)

RSSI is a measurement of power of the received radio signal.

The signal strength is affected by the environment, and is therefore unstable. To get a useful reading from it, the signal must be filtered. [11]

3.5 ESTIMOTE

Estimote is a company specializing in beacons. Their goal is to make beacons more accessible for developers. They have put a big emphasis on their API and developer community. Their focus is to make a level of abstraction above iBeacon to make it easy for developers to build consumer value. [12] [13]



Figure 8 Estimote [13]

3.6 S1000D SPECIFICATION

S1000D is an international specification for the procurement and production of technical publications. It is an XML specification for preparing, managing, and using equipment maintenance and operations information. It was initially developed by the Aerospace and Defence Industries Association of Europe (ASD) for use with military aircraft. The specification has since been modified for use with land, sea, and commercial equipment. S1000D is part of the S-Series of ILS specifications. [14] [15]

3.6.1 APPLICABILITY

Applicability provides the mechanism to identify the context for which a data module or parts of a data module is valid. This context is usually associated with the physical configuration of the Product but can include other aspects such as tool availability and environmental conditions. [16]

3.7 CORENA PINPOINT NEUTRAL PACKAGE

The data sets that the application can load is in the format Pinpoint Neutral Package. This is a package Flatirons Solutions has developed. Originally the technical documents are in a XML format. The neutral package reformats them to JSON files and a proper structure making it easier to interpret the data.

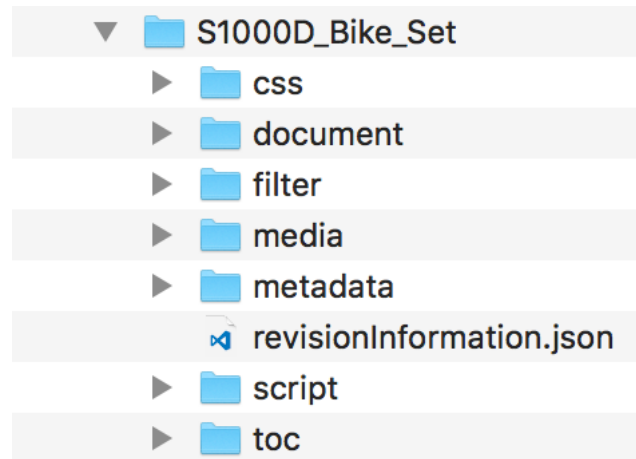


Figure 9 - Folder structure of CORENA Pinpoint Neutral Package

In Figure 9 you can see the folder structure. In the neutral package. We have developed a prototype that implements the table of contents(toc) from the toc folder. The toc navigates us to the right document in the document folder. Inside the documents, we have data that tells us what is applicable or not.

3.8 TEST DRIVEN DEVELOPMENT

Test-driven development (TDD) is an approach to development which combines test-first development where you write a test before you write just enough production code to fulfil that test and refactoring. [17] This ensures that the code you write is the minimum amount required to solve the problem.

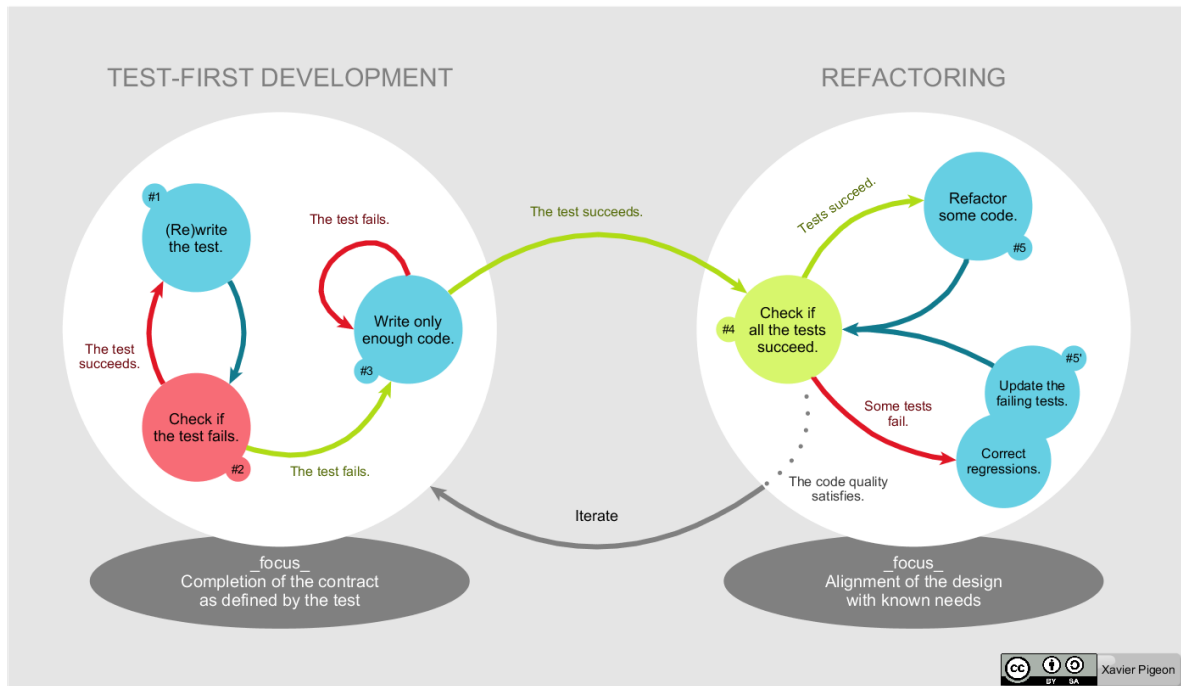


Figure 10 - TDD process [18]

There are five main steps to test-driven development [19]

1. Write a new test
2. Run all the tests and confirm that the new one fails
3. Make a small change to the code
4. Run all the tests and confirm that they all succeed
5. Refactor the code until code standard is good enough

3.9 UNIT TESTING

A unit test is an automated piece of code that invokes a unit of work in the system and then checks a single assumption about the behaviour of that unit of work. [20]

The purpose of a unit test is to uncover bugs and faults in the code early in the process. It also ensures that the units of the program remain functional even after changes, or warns about errors. [21]

3.9.1 QUNIT

QUnit is a JavaScript testing framework. It is designed to be simple to set up and give immediate detailed feedback.

3.10 AGILE MODELING AND DEVELOPMENT: SCRUM

We are using Scrum as our project model. Scrum gives us the possibility dividing the work into individual tasks, while keeping communication channels open. The project model lets us organize work for each sprint and assign that work to each person. We are then responsible for our own tasks that they get completed. Scrum enables us to react and adapt to requirement changes.

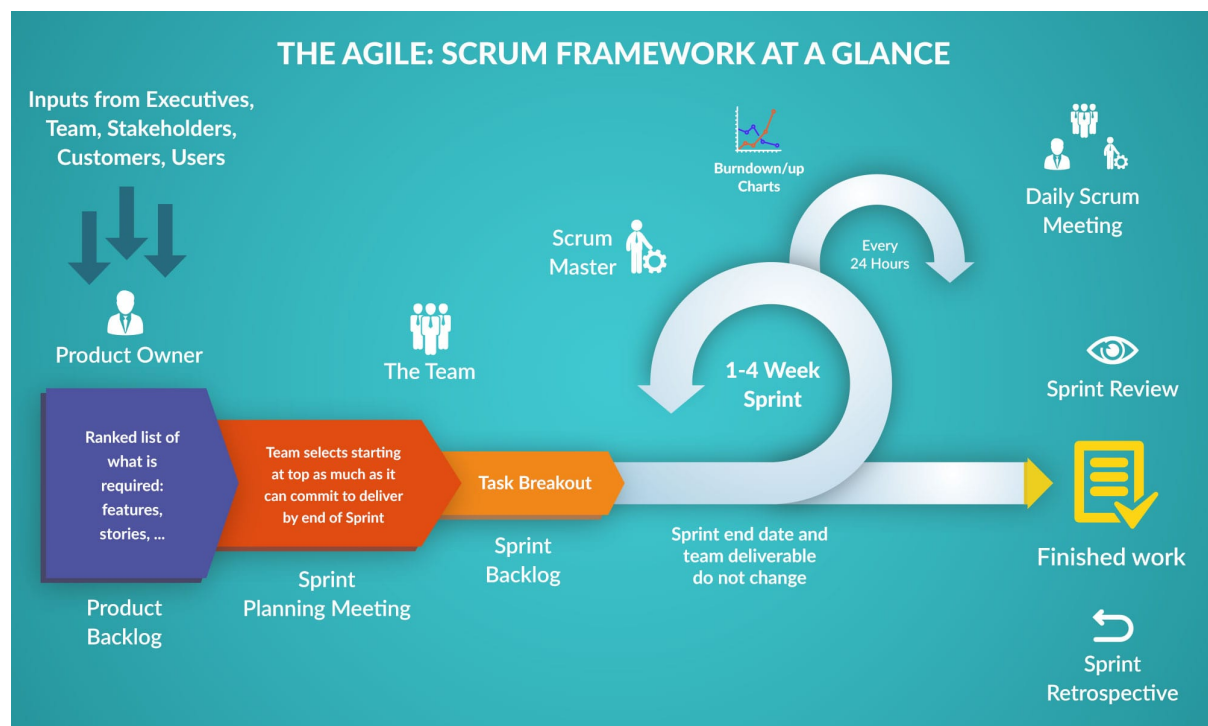


Figure 11 - Scrum Framework [22]

3.10.1 DAILY SCRUM MEETING

The daily Scrum meeting is at the start of every workday at a specific time, and is usually a short meeting. The meeting will be held regardless of members present. To ensure a short and concise meeting, all participants stand throughout the meeting. Each member of the team runs through these topics: what have been done, what must be done, problems that may hinder progress in a sprint. All technical and problem solving conversation is reserved for after the scrum meeting. The function of the scrum meeting is to quickly update the team about each members progress and problems, and establish actions to solve problems.

3.10.2 USER STORIES

A user story is a story explaining what a user does or needs to do in the users language. It captures the Who, What and Why of a requirement. Example of this is: As a User I want a Function so that Why. In our process model, we are using user stories for our requirements. If a user story is too big to be done in one sprint we use an Epic story instead. Epic stories are stories that can go over multiple sprints. If we find it necessary, we break user stories down into tasks or even further into subtasks. [8]

3.10.3 PRODUCT BACKLOG

The product backlog consists of an ordered list of issues. The issues can be features, bug fixes, non-functional requirements, etc. This includes everything that must be done to successfully deliver a viable product. The product backlog is not the same as a sprint backlog. This backlog is for the whole project while the sprint backlog is only for the active sprint.

3.10.4 SPRINT

A sprint is a set of tasks that should be resolved within a period. Generally, a sprint is between 1-4 weeks long.

3.10.4.1 SPRINT BACKLOG

The sprint backlog is selected from a subset of the product backlog, and does not usually change during a sprint. Created during sprint planning, the sprint goal helps give meaning to the stories and sprint goal. Stories should be decomposed into manageable subtasks.

During the sprint, we have an active sprint backlog. All our tasks are here and we can freely assign them to each team member. The goal of the sprint is to close all tasks in the backlog during the active sprint.

3.10.4.2 SPRINT PLANNING

Preceding every sprint, the team discusses the sprint goal and which subtasks to include. This means that the team evaluates how many story points can be completed, and select the highest priority tasks to complete.

3.10.4.3 SPRINT REVIEW

Following each sprint, the team demonstrates the product to the product owner and interested stakeholders. The result of the sprint is in the form of a potentially shippable product increment. The feedback from the meeting is used to create new backlog items, and the backlog is reordered according to importance.

3.10.4.4 SPRINT RETROSPECTIVE

After the sprint review, the team gathers for a sprint retrospective. This is a walkthrough of the sprint period. The team discuss which elements of the sprint went well, what could have been done better and impediments. This gives us the insight we need to improve each sprint by identifying key problems every sprint.

You will find our sprint retrospectives in appendix C

3.10.4.5 BURNDOWN CHART

A burndown chart is a graphical representation of work left to do versus time. It shows the actual and estimated amount of work to be done in a sprint and helps a team to manage its progress throughout the iterations and predicting when all the work will be completed. It is useful to estimate likelihood of achieving the sprint goal.

The horizontal x-axis in a Burndown Chart indicates time, and the vertical y-axis indicates story points. Below is an example of a burndown chart from one of our sprints.

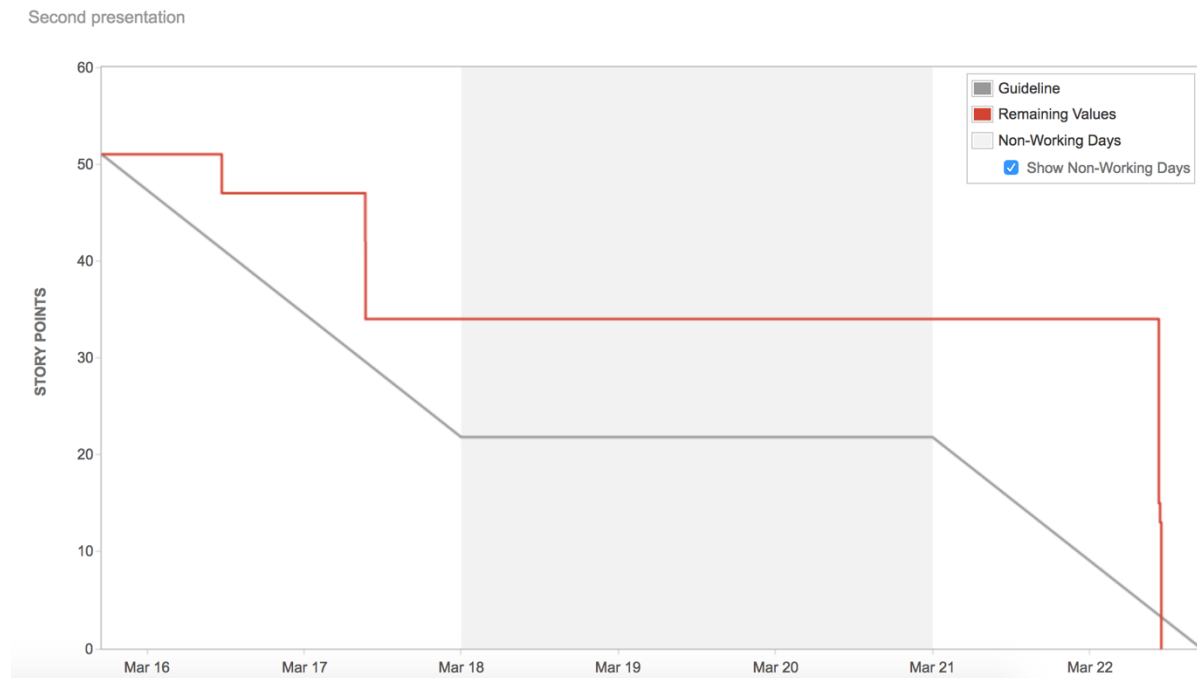


Figure 12 – Burndown chart from Sprint 8

3.11 RISK

Risk analysis is the systematic process to estimate the level of risk for identified and approved risks.

The engineering risk management process involves five components. In a project *management plan*, each of these could have their own heading. The components are: [23]

- Planning for risk
- Identifying risks
- Analysing risks
- Developing risk response *strategies*
- Monitoring and controlling risks



3.12 PROTOTYPING

One of the fundamentals of prototyping is that it is generative. This means you will generate hundreds of ideas that could be brilliant. Most of them will not be, but some will be. As a generative process, prototyping often leads to innovation and saving of costs and time. It helps you get the ideas out of your head and into something tangible. [24]

It is possible to design a prototype and then test that prototype and get valuable feedback to see if the idea has any value.

4 METHODOLOGY

In any project, it is important to use a process model. It is aimed at providing guidance for developers. For our project, we have decided to use Scrum. When we set out to find a process model we looked at different models including Unified Process. We had a discussion if we would go for Unified Process or Scrum. We decided to use scrum since our external supervisors were familiar with the model, and we needed an agile project model to cope with rapid development.

4.1 HOW SCRUM HAS BEEN USED

We had a daily Scrum meeting to go through what we did the day before and what we were going to do that day. This was a good way to plan the progress of the work that day.

The duration of the sprints was one week. Every Wednesday we had sprint demo, sprint review and sprint planning. This sprint demonstration was a good way for Flatirons Solution to see what we were making and how we were progressing. This opened good channels for communications with Flatirons Solutions.

We chose Jira to keep track of our Scrum activities. This made it easy to update the scrum backlog and sprint goal.

4.2 GROUP DYNAMICS

The most important characteristics of the bachelor group is the diversity of its members. Each member has different strengths which benefits the team. The group has a consensus approach to all challenges and manages excellent in problem solving. The bachelor group has encountered and overcome every impediment with ease. This is due to high level of communication and matureness within the group. With the team in focus all members contribute to make a prosperous development to ensure an effective team and a perfect product.

4.3 COMMUNICATION WITH FLATIRONS SOLUTIONS

Being at Flatirons Solutions made it easy to get up from the desk and ask any of the employees, external supervisors or the director of R&D Flatirons Solutions Norway if we encountered any challenges or needed guidance. We used Skype as a channel to communicate with Flatirons Solutions to arrange meetings, or ask questions.

4.4 MEETINGS WITH INTERNAL SUPERVISOR

HSN changed their mail system at the start of the semester. This made it difficult to get in touch with our internal supervisor the first two weeks. By investigating the matter, we finally found a working communication and the location of the internal supervisor. After that we established an excellent cooperation. In the latter part of our project we had direct contact with internal supervisor and conducted most of the documentation work at the school premises.

4.5 SEATING ARRANGEMENT

We had two office cubicles and one conference room at our disposal at Flatirons Solutions premises. Usually we used the cubicles where there were room for two persons at the desk. The cubicles were right next to each other so we frequently got up and talked to the two other members of the bachelor group. We also had a monthly change of seating arrangements to improve group dynamics. When we had Scrum review and Scrum planning we used the conference room. For the daily Scrum meetings, we used one of the office cubicles or meeting rooms. We spent most of our project at Flatirons Solutions. We spent the final two weeks at the school to discuss the documentation with our internal supervisor

4.6 RISK

Our risk document was continuously updated as risks developed.

4.6.1 IDENTIFYING RISKS

Project risks should be examined to a level of detail that permits an evaluator to understand the significance of the risk and its causes, and to potentially examine the root causes. Some categories we have considered [10]

- **Hardware Resources** - Hardware fails or is not available
- **Software Resources** - Software fails, have unexpected bugs or is not available
- **Human Resources** - Project staff is not available, or lose availability
- **Risk for the user of the software** - In case bugs or software fails for the customer

4.6.2 DEVELOPING RISK RESPONSE STRATEGIES

Each risk we identify had a planned risk response. It is important to plan for possible risks. By having a risk response strategy for every risk we identified, we were be more prepared.

4.6.3 MONITORING AND CONTROLLING RISKS

Monitoring and controlling risks was done continuously. In our daily Scrum meetings, any concerns for risk was mentioned. Risks were evaluated in the sprint planning meetings and in the retrospectives.

4.6.4 RISK MATRIX

We used a risk matrix to determine the severity of the risk. It works by figuring out the likelihood of the risk to occur and the severity if it occurs.

COLORS AND NUMBERS EXPLAINED

| | | | | | | |
|------------------------------------------------|---|----|----|----|----|----|
| L i k e l y h o o d | 5 | 10 | 16 | 20 | 23 | 25 |
| | 4 | 7 | 13 | 18 | 22 | 24 |
| | 3 | 4 | 9 | 15 | 19 | 21 |
| | 2 | 2 | 6 | 11 | 14 | 17 |
| | 1 | 1 | 3 | 5 | 8 | 12 |
| | | 1 | 2 | 3 | 4 | 5 |
| Consequence | | | | | | |

Figure 13 - Risk Matrix, based on risk matrix from [25]

The risk matrix is colour coded using red for high priority risks and green for low priority. Numbers 10 and under are considered low risk and do not need much attention. Any risk ranked between 11 and 19 should be followed up closely. A mitigation plan should also be made. Risks above 20 should be monitored very closely and we should work to reduce the risk at all times [11]

HARDWARE RISKS

| Risk identification | Risk analysis | Risk Response | Risk Rank |
|---------------------|------------------------------------------------------------------|-------------------------------------------------------------------------------------|-----------|
| Faulty beacon | Small chance of getting a faulty beacon or the beacon failing | We always have multiple beacons. In case we receive a faulty. Contact manufacturer. | 5 |
| Faulty computer | Small chance that any of our available computers can fail | Have multiple computers that can do the tasks that we need done | 3 |
| Faulty phone | Small chance that any of our available phones can fail | Have multiple phones that can do the tasks that we need done | 1 |
| Inaccurate beacon | High chance that beacon signal strength will fluctuate over time | Mitigate errors through software correction | 18 |

SOFTWARE RISK

| Risk identification | Risk Analysis | Risk Response | Risk Rank |
|---------------------|----------------------------------------------------------------|-----------------------------------------------------------------------------------------------|-----------|
| Loss of data | Low chance of losing documents or source code. | Everything is stored on our computers and in a cloud server. This ensures we do not lose data | 8 |
| Bugs in frame-works | Possibility of bugs in the framework we use that can affect us | If we find a bug that affects us, add it to the backlog as an issue | 16 |

HUMAN RESOURCES

| Risk Identification | Risk Analysis | Risk Response | Risk Rank |
|---------------------|----------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Absence | High chance of some absence throughout the project | Persons that are absent need to inform the group. When they get back they need to catch up with their work. Longer days may be required | 10 |
| Long Absence | Low chance of long absence | For longer absence, the group will have to cover the tasks of the person that is away. More work is expected | 17 |

RISK FOR USER OF APPLICATION

| Risk Identification | Risk Analysis | Risk Response | Risk Rank |
|---------------------|---------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|-----------|
| Wrong information | There is a possibility of the application giving the user the wrong information | Have a clearly defined document so the user easily understands if he is in the wrong document | 14 |
| Faulty beacon | Possibility of a faulty beacon | Not much that can be done about that. User will have ability to see beacon status. | 3 |

4.7 TESTING

Tests are important to make sure we deliver a product that complies with the requirements that have been set. To make sure we do test properly we are using (Test Driven Development).

As an example, to test an applic statement, we constructed classes that follows the structure as specified in S1000D, and delegate the content down to the lowest level asserts. These asserts take care of comparing the values of the applic types and simplify the problem to comparing true - false values.

For more details about the tests performed, go to appendix E

4.8 NATIVESCRIPT APPLICATION ARCHITECTURE

Our application is built using NativeScript with Angular, which is built on a component-based architecture.

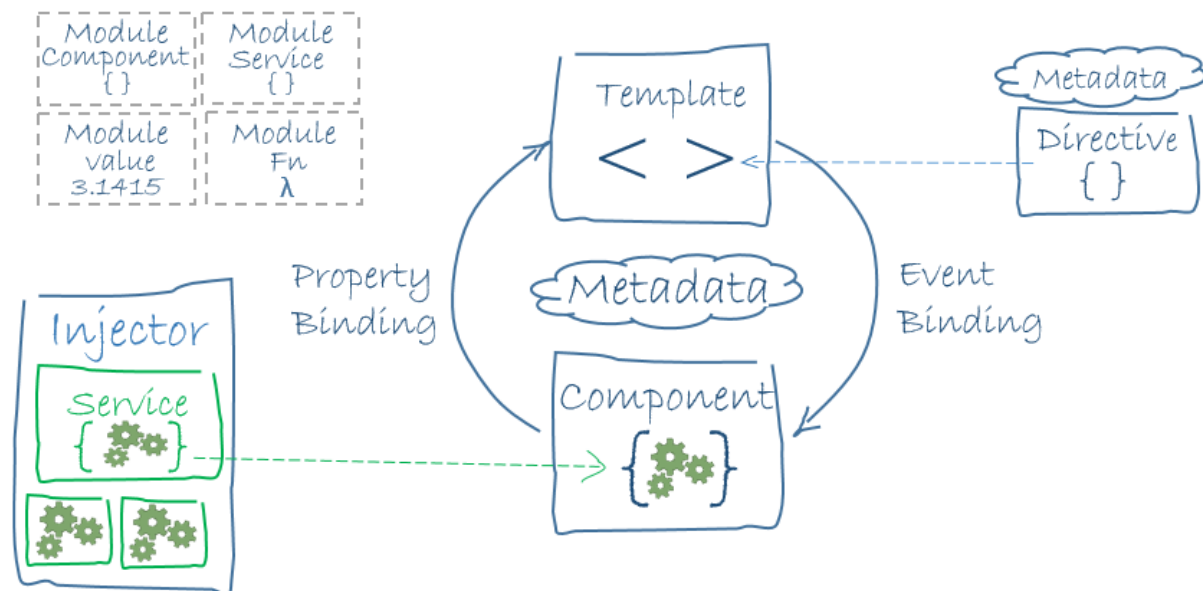


Figure 14 - Angular architecture [25]

4.8.1 COMPONENT

The component is the main building block the application. This is what controls the view of the application.

Component classes should be small, and perform trivial tasks. Non-trivial operations should be delegated to services.

4.8.2 TEMPLATE

Templates defines how a component should be displayed. A template communicates with its component through data bindings.

4.8.3 SERVICE

Service is a broad category encompassing any value, function, or feature that your application needs.

4.9 STYLE GUIDE

The team followed the official style guide for Angular when developing the project. [26].

We have also followed java code conventions for indentation and code formatting.

<http://www.oracle.com/technetwork/java/javase/documentation/codeconvtoc-136057.html>

4.10 JSDOC

The team used JSDoc to document the code base. The wiki page is provided on the CD.

5 REQUIREMENTS

During our project, Flatirons Solutions have given us certain requirements that we need to meet:

Original Requirements:

- Cross platform
- Use beacons to filter applicability
- Use a mock data set

Added Requirements during project:

- Use DC9 data set
- The ability to swipe through documents

We have broken the requirements into user stories and given them all acceptance criterias. During the 5 months of the project we have created a massive number of stories to be done. We have used the backlog board for both development and documentation and estimated that including all of them would be overwhelming in this section, we have listed the stories that is the most significant. For an overview of all our user stories please look at the attached user stories document.



Figure 15 DC9 Airplane [27]

5.1 FUNCTIONAL

Functional requirements define a function of a system. A function is described as a set of inputs, behaviours and outputs. We had a few functional requirements given by Flatirons Solutions.

5.1.1 SWIPE

As a technician

I want to be able to swipe through documents that are under the same toc node
so that I can more easily navigate through documents

ACCEPTANCE CRITERIA

Given CORENA pinpoint document is open

When the user swipes to left

Then the next document as listed in the toc node opens

Given CORENA pinpoint document is open

When the user swipes to right

Then the previous document as listed in the toc node opens

5.1.2 DETECT BEACONS

As a user

I want to be able to detect nearby beacons
so I can communicate and retrieve data

ACCEPTANCE CRITERIA

Given Beacon is nearby

And Bluetooth is enabled

And Permissions is set

When Application starts

Then Application detects beacon

5.1.3 FILTER HTML COMPONENTS

As a technician

I want to be able to see only the relevant parts of a document
so that I can more easily find the right documentation

ACCEPTANCE CRITERIA

Given An applicability filter specified

When An HTML document is opened

Then Only applicable elements is shown

5.1.4 BEACON LIST UI

As a technician

I want to be able to see a list of nearby beacon groups
so I can quickly access the information of that group

ACCEPTANCE CRITERIA

Given Beacons are active

And Beacons have beacon group data

And Device has Bluetooth enabled

When The device in range of a beacon group

Then beacon group is shown on the device

Given Beacon group(s) is displayed on the device

When The beacon group is selected

Then The applicability filter of that group is displayed

And The beacons of the group is displayed

Given A beacon group is selected

When The applicability filter of that group is clicked

Then The applicability filter is applied

5.2 NON-FUNCTIONAL

A non-functional requirement specifies criteria that can be used to judge the operation of a system rather than the behaviour.

5.2.1 DC9 DATA SET

As a technician

I want to be able to view the DC9 data set
so that I can find applicable documents

ACCEPTANCE CRITERIA

Given DC9 data set is present in the app

When DC9 data set is selected

Then DC9 data set is shown

5.2.2 CROSS PLATFORM

As a user

I want to be able to use the application on both android and iOS
so that I can have the ability to choose platform.

ACCEPTANCE CRITERIA

Given Estimote plugin installed

When call to EstimoteSDK

Then the desired action is performed

6 MODELING THE PRODUCT

This chapter contains the diagrams from use case of the application to the software architecture. A specific use case or component can be traced through the diagrams by looking the colour of the background.

Chapter 7 goes into details of how the architecture is implemented. This includes class diagrams, and sequence diagrams showing details.

Following component-based architecture, the architecture design in 6.3 is modular, and components can easily be added or replaced.

6.1 USE CASE

To get an overall idea of how the user should interact with our demo application we have created a use case diagram.

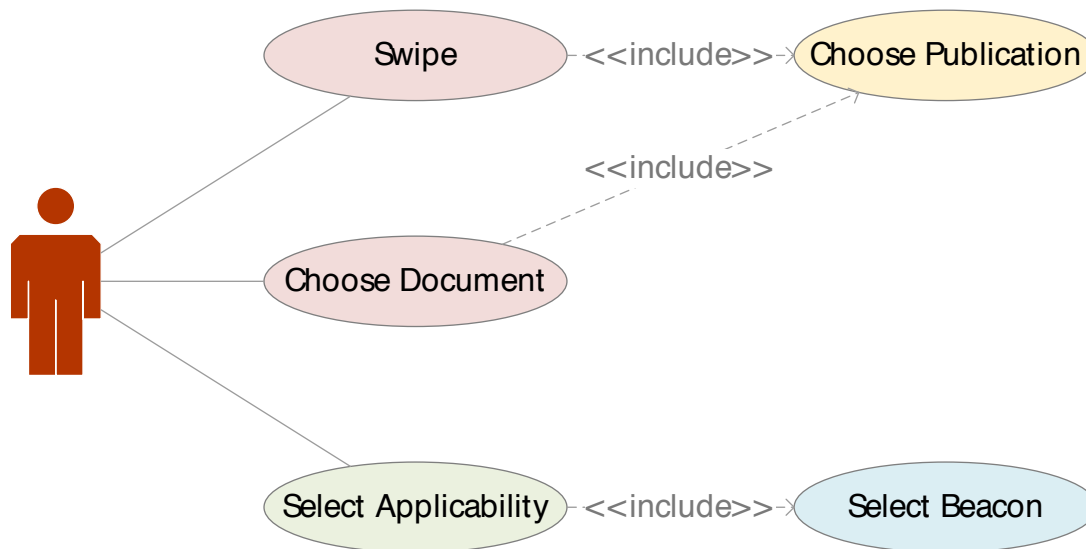


Figure 16 - Use Case model

CHOOSE PUBLICATION

Select publication to display. The publication contains the list of documents used when selecting document.

SWIPE

Go to the next or previous document when viewing a document.

CHOOSE DOCUMENT

Select a document from the table of contents.

SELECT APPLICABILITY

Select which applicability statement should be used to filter the documents.

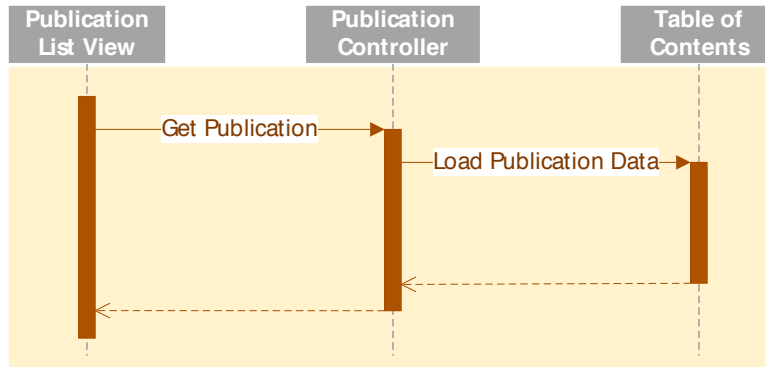
SELECT BEACON

The ability to select beacons that have applicability to filter the documents and table of contents

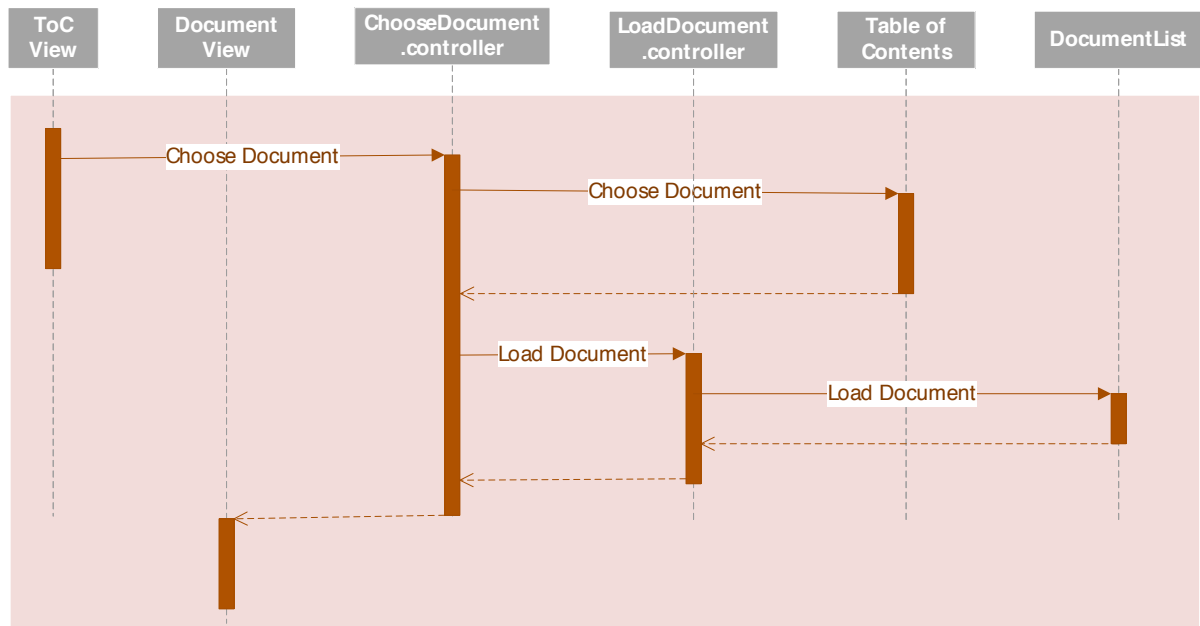
6.2 SEQUENCE DIAGRAM

The sequence diagrams in this section shows the sequence when the user interacts with the application. The sequence diagrams have coloured backgrounds to represent the correct use case in the use case model from Figure 16

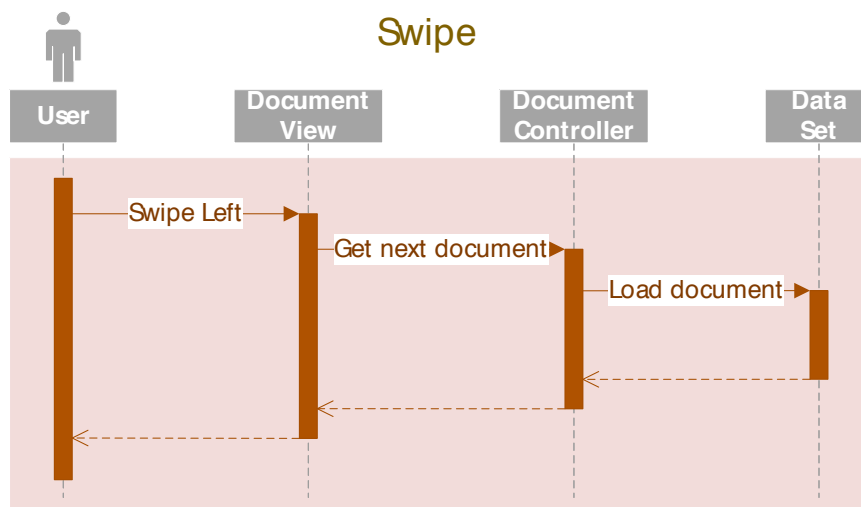
6.2.1 CHOOSE PUBLICATION



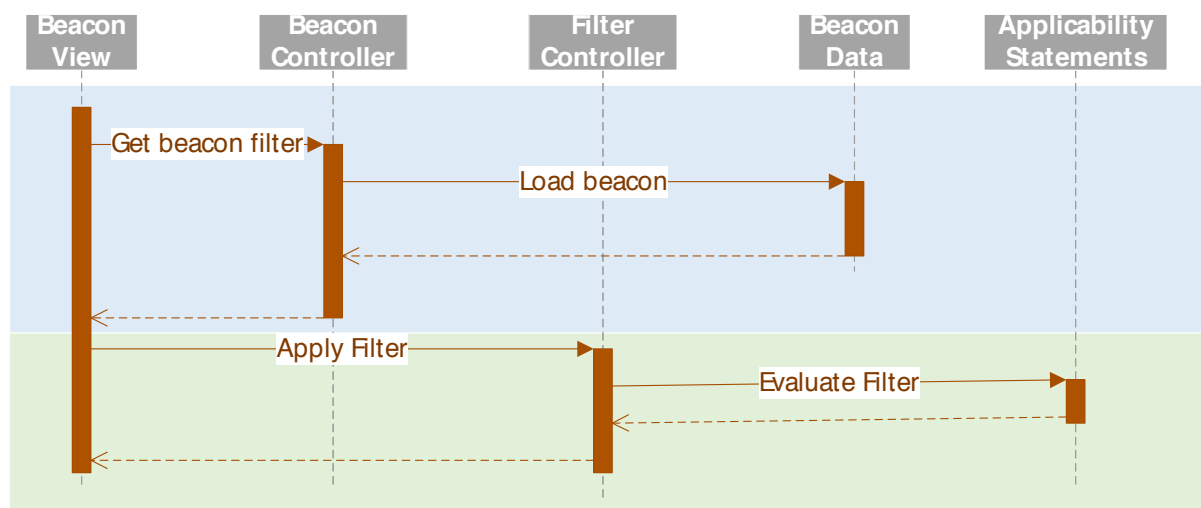
6.2.2 CHOOSE DOCUMENT



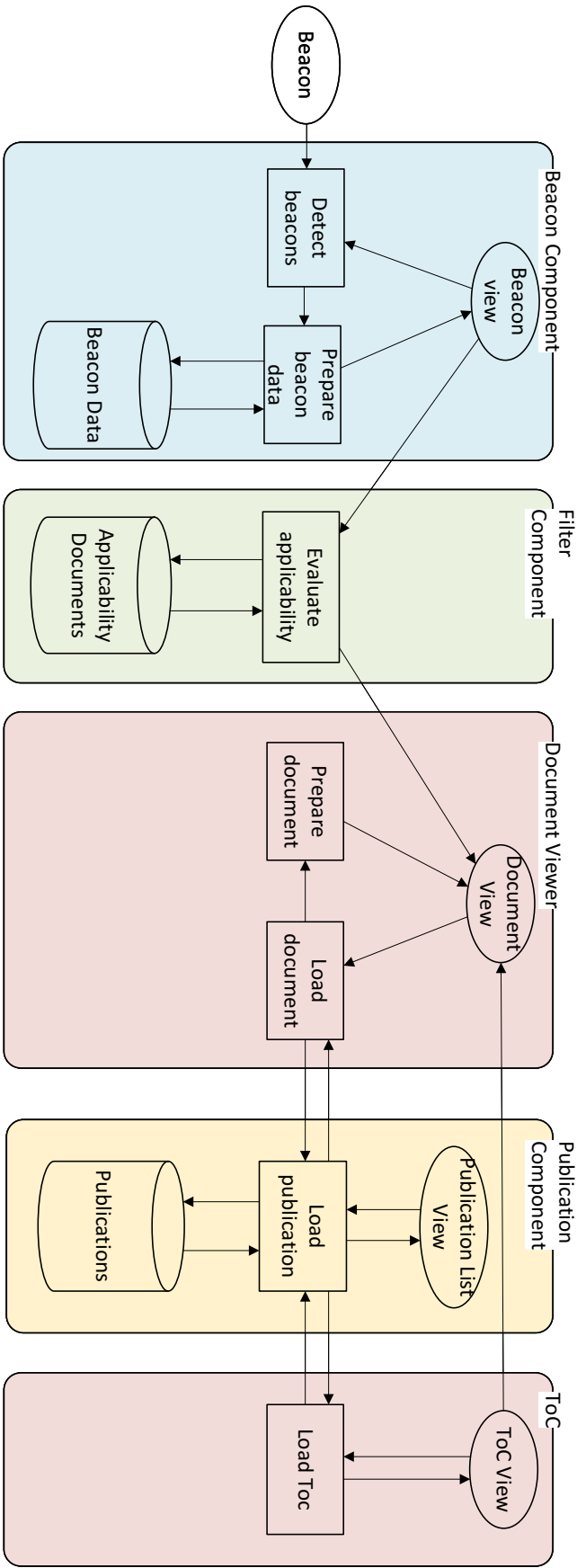
6.2.3 SWIPE



6.2.4 SELECT BEACON

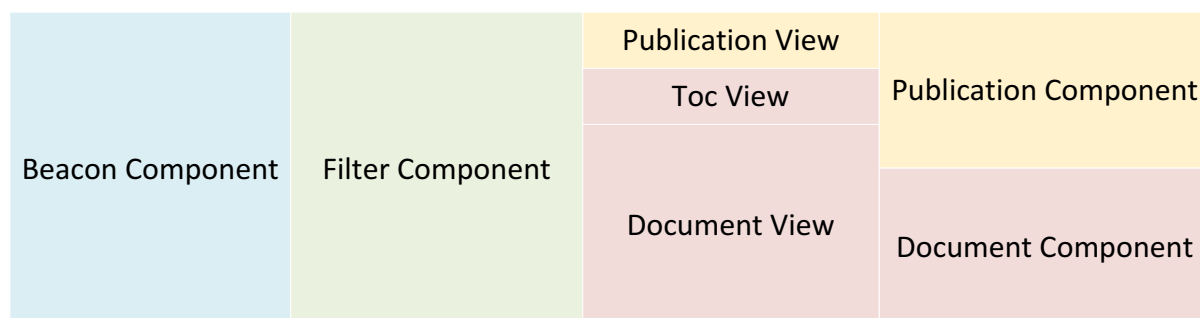


6.3 SOFTWARE ARCHITECTURE



7 THE IMPLEMENTATION

This chapter introduces the implementation in detail. Every subchapter starts with a class diagram with colours indicating what part of the architecture it implements.



7.1 PUBLICATION MANAGER

The publication manager takes care of all access to a publication and its belonging documents. It is meant as a layer of abstraction to avoid direct contact with a pinpoint neutral package and its content.

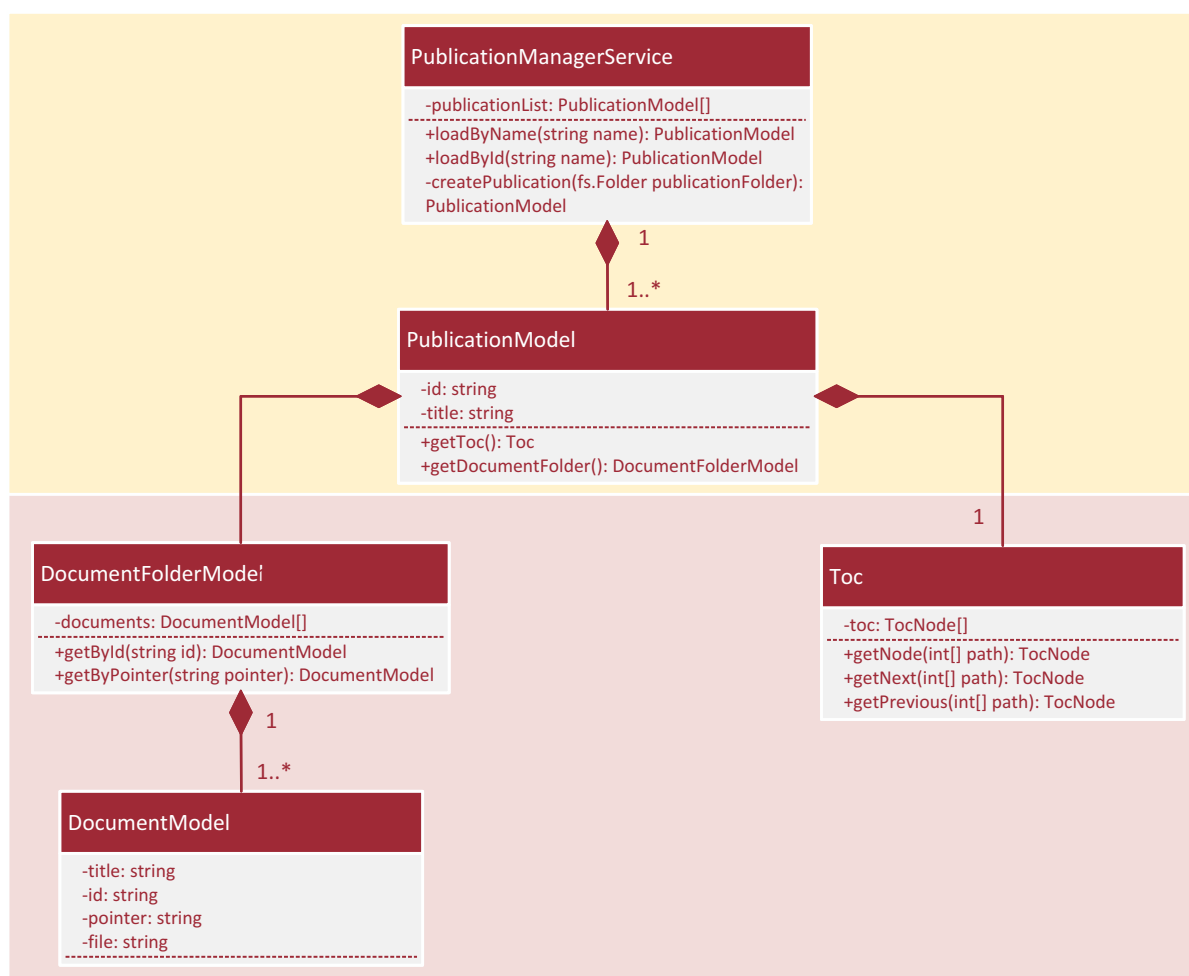
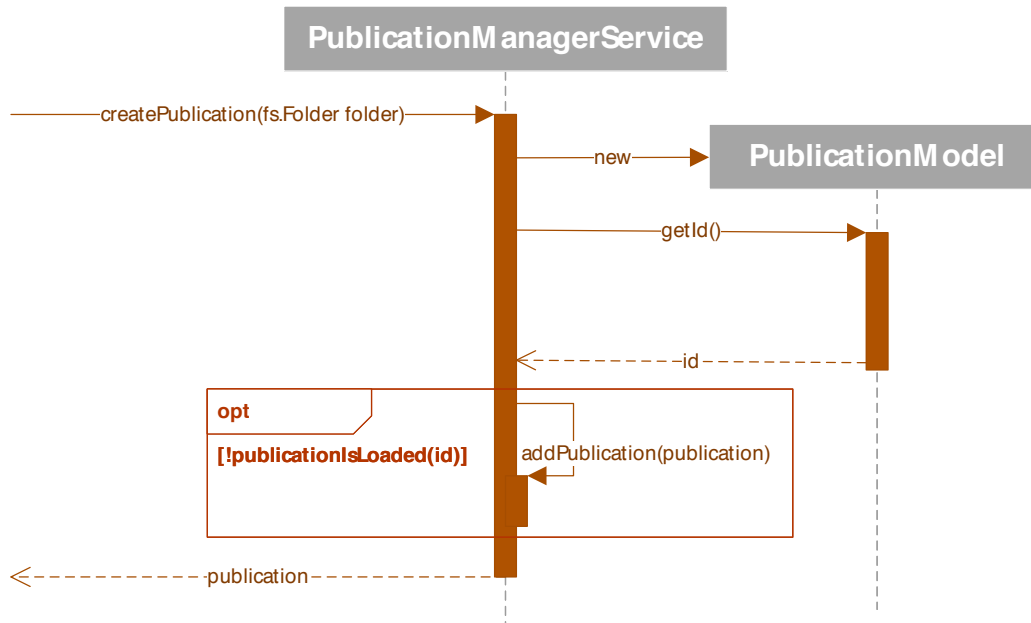


Figure 17 Publication class diagram

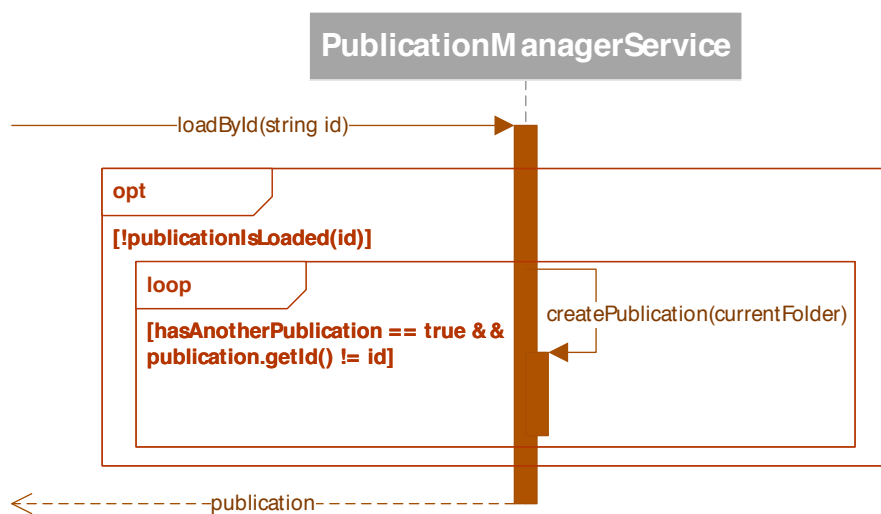
7.1.1 CREATING A PUBLICATION

7.1.1.1 CREATE PUBLICATION USING PUBLICATION FOLDER

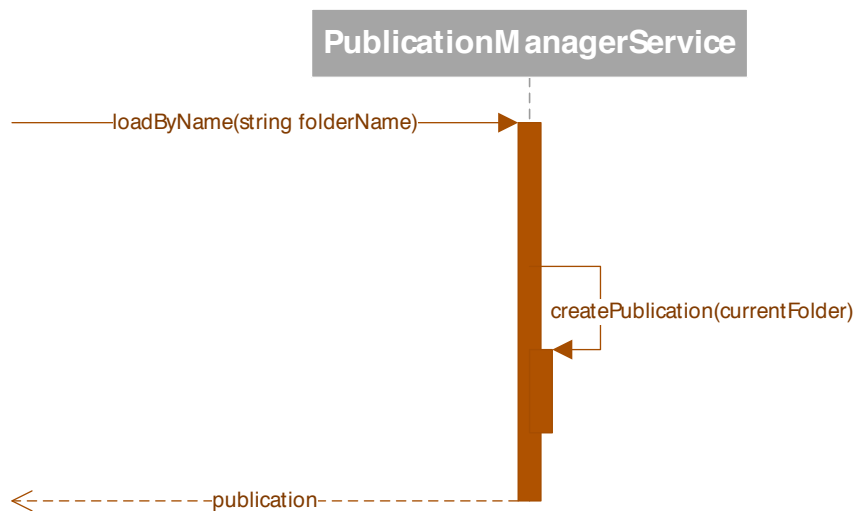


7.1.2 LOADING A PUBLICATION

7.1.2.1 CREATE PUBLICATION USING PUBLICATION ID



7.1.2.2 CREATE PUBLICATION USING NAME



The publication model consists two parts:

- Documents
- Table of contents

7.1.3 DOCUMENTS

For each publication, we need to know the location of the folder containing the documents. The location will vary based on each publication. In Figure 18 you can see the sequence required to create a documentFolder.

7.1.3.1 GET DOCUMENT FOLDER

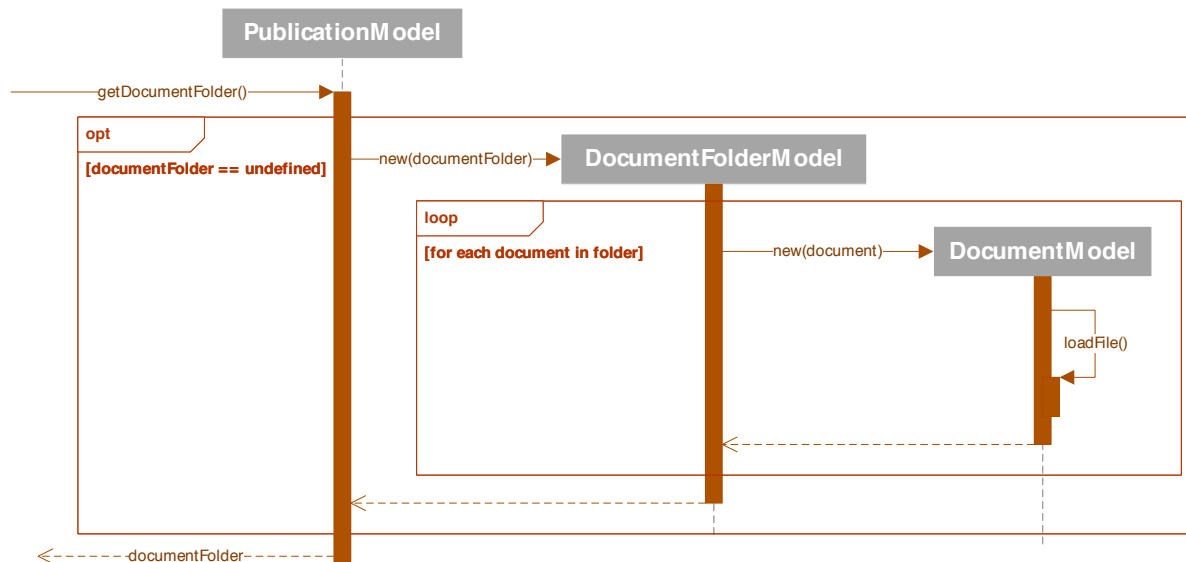


Figure 18 - Creating a documentFolder

7.1.4 TABLE OF CONTENTS

Each publication has their own table of contents file located in the pinpoint neutral package.

7.1.4.1 LOAD TABLE OF CONTENTS

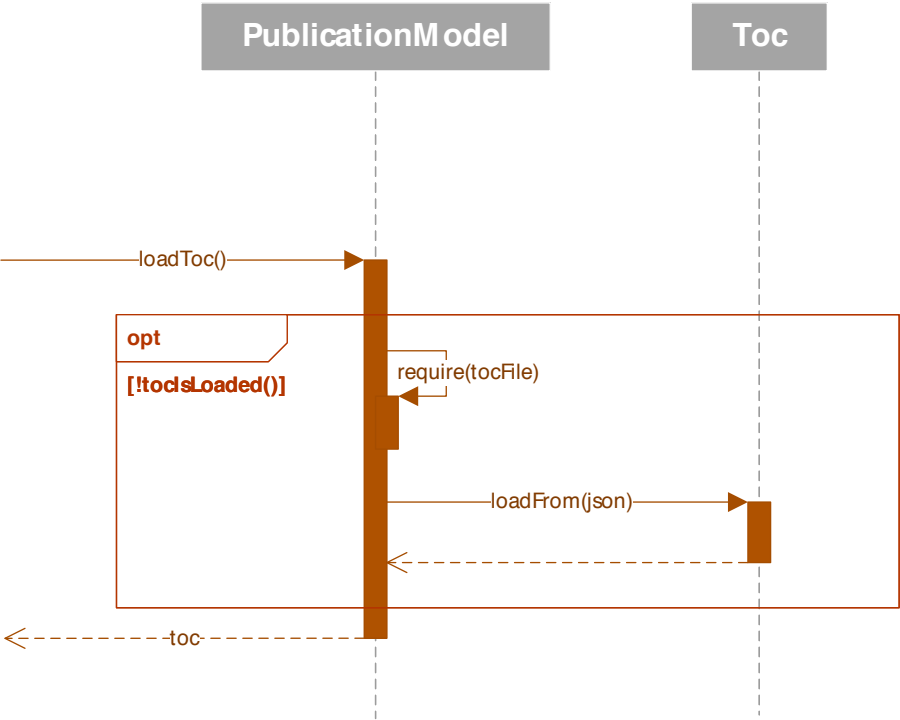


Figure 19 - Sequence diagram loading table of contents

Figure 19 - Sequence diagram loading table of contentsFigure 19 shows the sequence needed to load the table of contents for the publication. After we call the loadToc() the PublicationModel fetches the correct table of contents file depending on which publication we requests. The contents of the file gets returned as the toc.

7.2 FILTER COMPONENT

Applicability in the pinpoint publication is constructed as described by S1000D, but uses JSON instead of XML. The overall structure of the Applic element is the same and is wrapped in a Statement element.

The statement element contains the applic statement with other metadata, including display text and a filter id. The filter id is a hash of the applic statement, and is unique to the statement.

7.2.1 S1000D EVALUATING FILTER STATEMENTS

APPLIC

While S1000D 4.1 specifies that an <Applic> element must contain at least displayText, assert, evaluate or expression, the applic elements we work with can contain either an assert or evaluate, but can be left empty.

From testing the children of an applic, the result will say if the content is valid or not based on the result.

EVALUATE

The <Evaluate> element groups multiple tests, and specifies the logical operator that should be applied to the group

The testable elements that can be used are <Evaluate> and <Assert>. To process the statement, all child elements are evaluated and returns either true or false. The logical operator is applied, and a true or false is returned. If the content is valid, true is returned. False means that the content is not valid.

ASSERT

The <Assert> element tells you what values of the identifier something is applicable. An assert can be used to represent different types, and ranges. To represent multiple values, they may be separated by a vertical bar "|". To represent a range of values, separate the values with a tilde "~"

```
applicPropertyValues="2|5"applicPropertyValues="5~7"
```

These can be mixed to select multiple ranges or values

```
applicPropertyValues="2|5~7|9~13"
```

The result of the applic evaluate must be true for the settings is applicable

COMPLETE EXAMPLE

Following is a complete example of an applic statement from S1000D

```
<applic>
<evaluate andOr="and">
<assert applicPropertyIdent="version" applicPropertyType="prodattr" applicPropertyValues="mk1|mk9"/>
<assert applicPropertyIdent="model" applicPropertyType="condition" applicPropertyValues="Brook trekker"/>
</evaluate> </applic>
```

7.2.1.1 CREATING THE STATEMENT TREE

The objects in the statement tree are instantiated by supplying the top Statement object with an applicStatement document. The data will be read and traversed in order to create the correct structure with the correct logic.

To determine how to evaluate the AssertLists, both functions are defined in a JSON object with the keys 'and' and 'or'. The key is selected from the 'evalute' property of the applicList element. (Note: We did not misspell evalute)

The value of the 'evalute' property is used as a key to set the evaluate callback.

7.2.2 FILTER COMPONENT CLASS DIAGRAM

The filter component part of the architecture takes care of evaluating and applying filter statements.

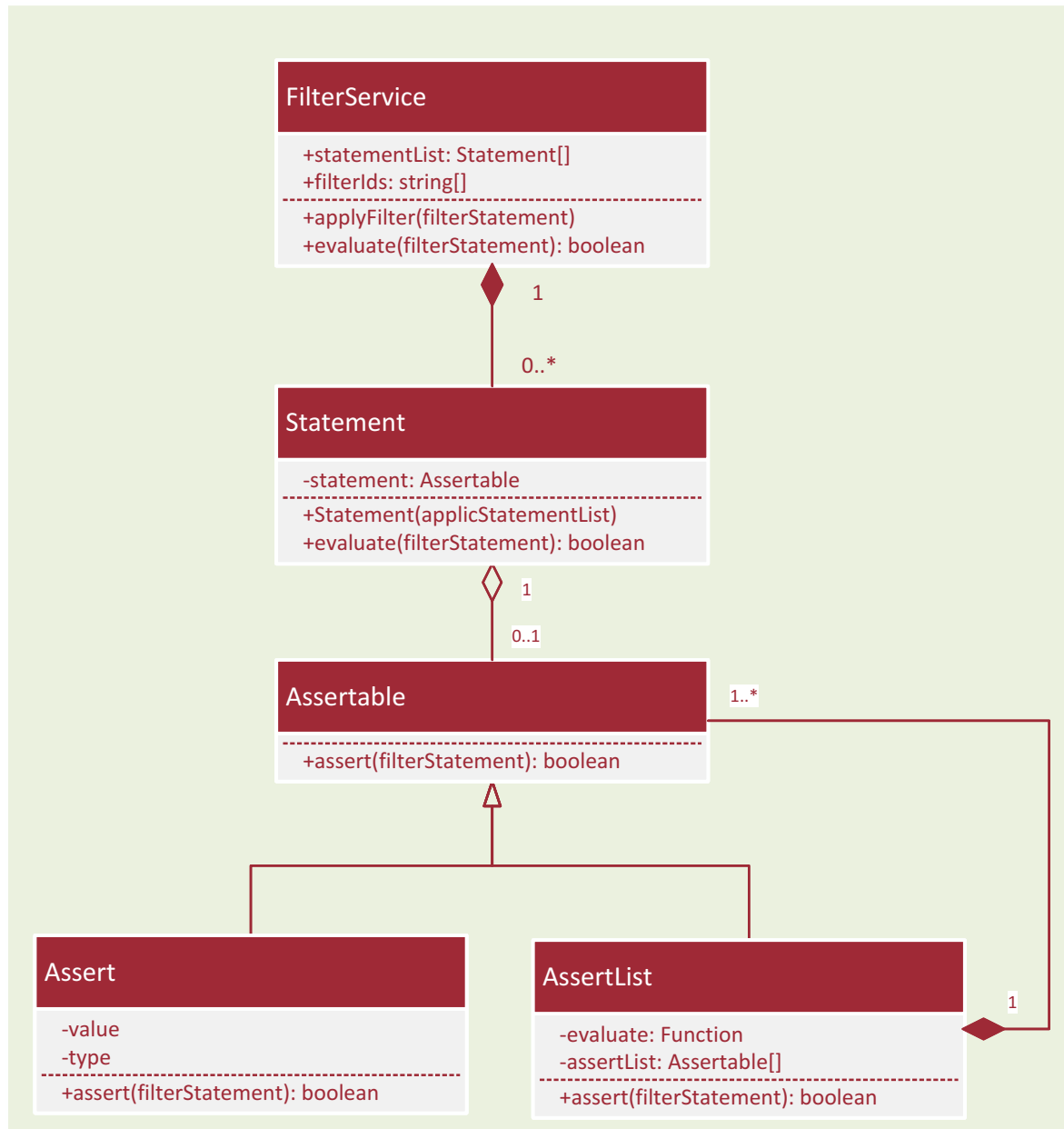
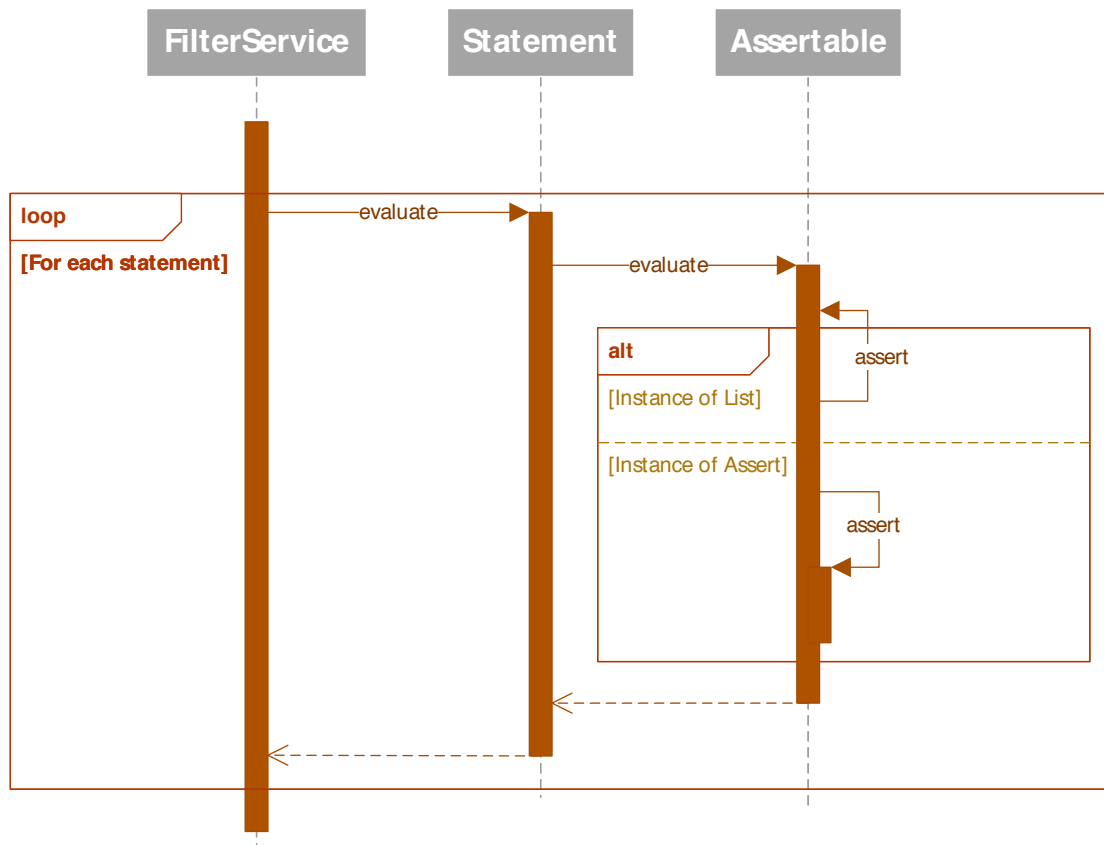
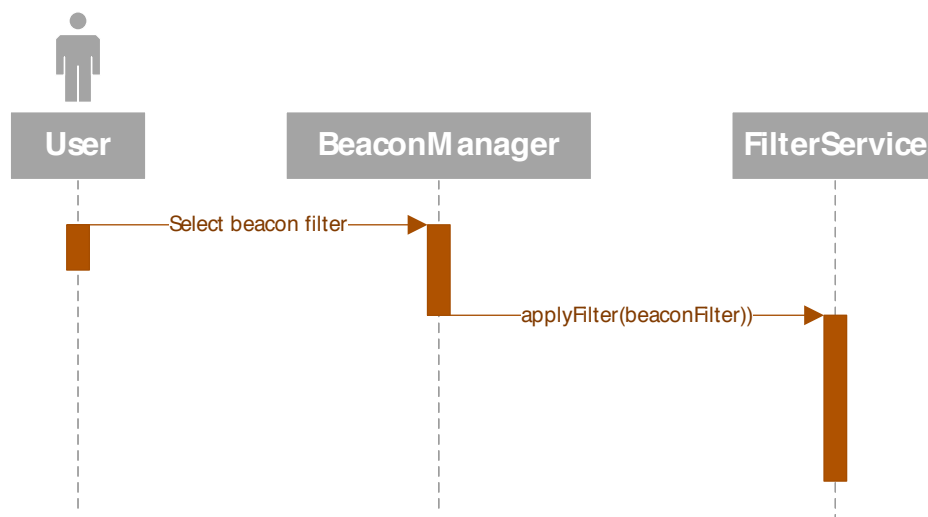


Figure 20 - Class diagram for evaluating filter statements

7.2.3 EVALUATING FILTER STATEMENT

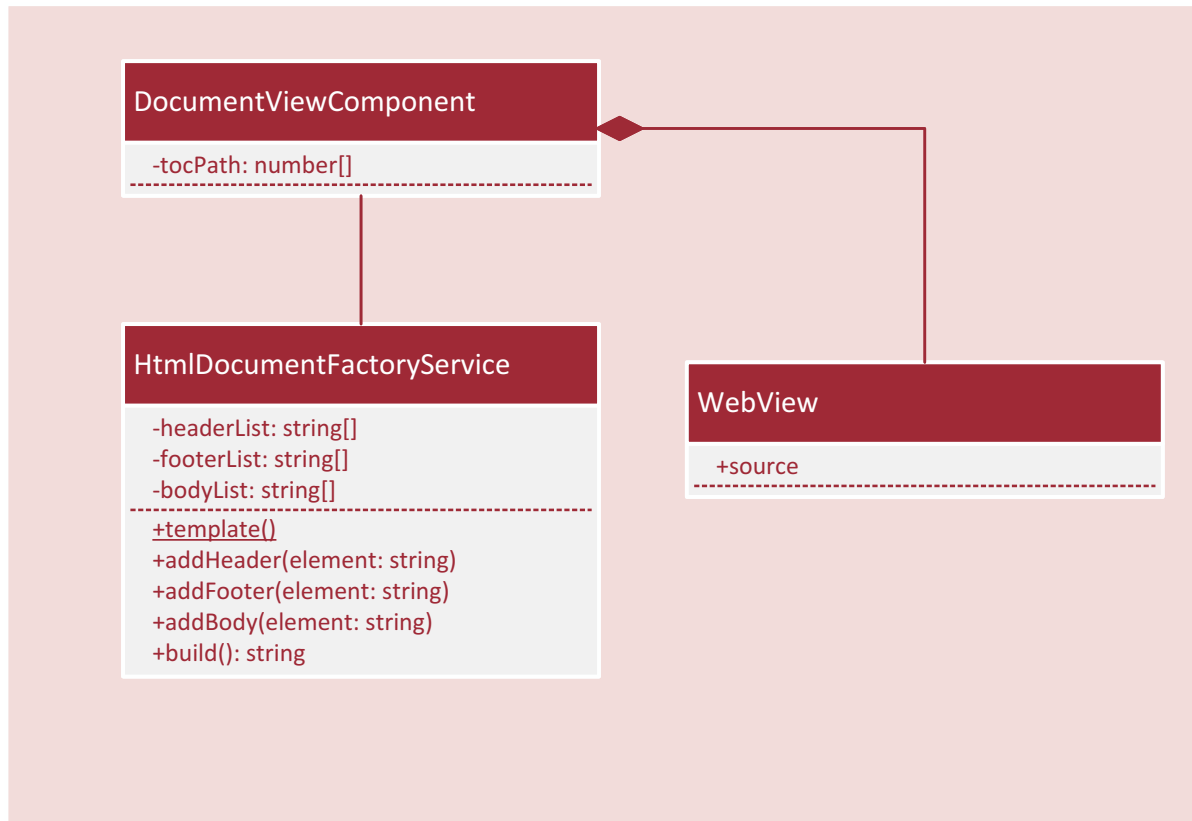


7.2.4 APPLYING FILTER STATEMENT

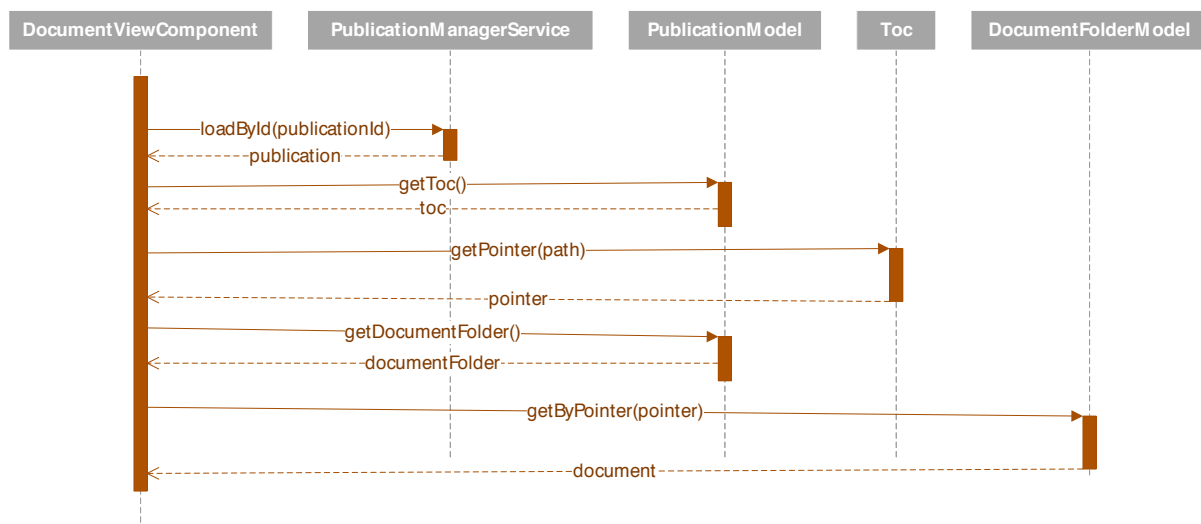


7.3 DOCUMENT VIEWER

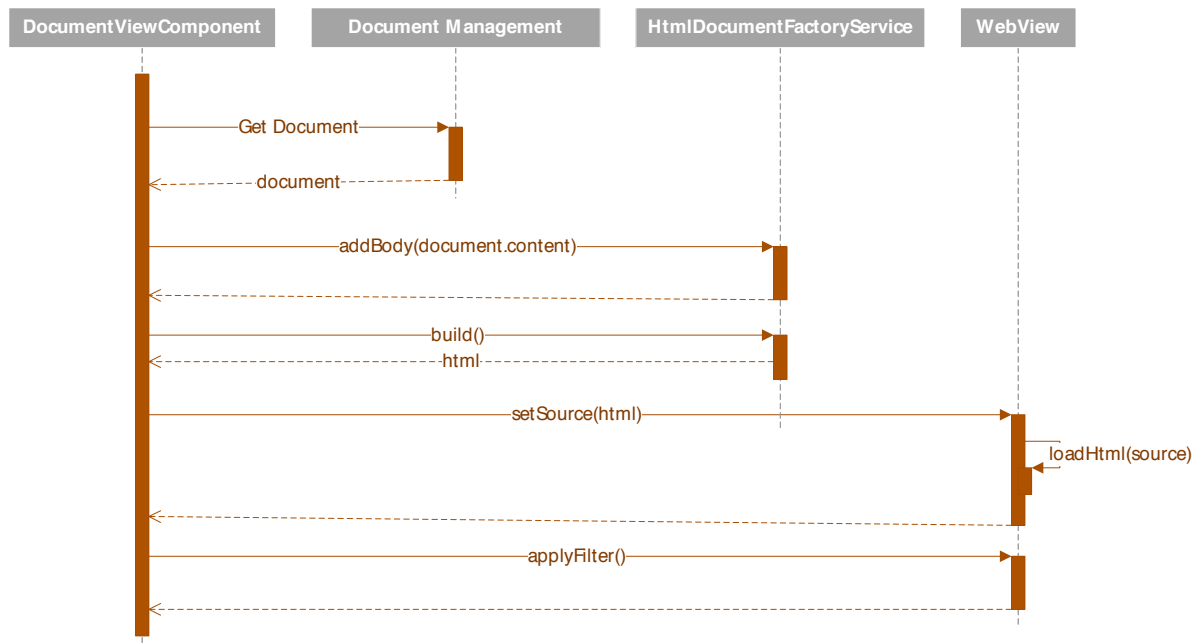
Flatirons Solutions has provided us with data sets to test our application. These contain HTML documents that will be displayed in our application. To do this we used the WebView interface provided by NativeScript.



7.3.1 LOADING DOCUMENT FROM PUBLICATION



7.3.2 DISPLAYING DOCUMENT IN WEBVIEW



7.3.3 WEBVIEW

To display the S1000D documents we use a WebViewer. This is to display the HTML and run the scripts that the documents contain. The WebView works in a similar way to a regular web browser, applying CSS and running JavaScript in addition to displaying the HTML. A challenge using the WebView is that the context is separate from the rest of the application. Accessing the DOM elements from the app was not a trivial task.

```

{
  "content": "HTML CONTENT",
  "id": "S1000DBIKE-AAA-D00-00-00-00AA-00PA-D",
  "title": "Mountain bicycle – Products cross-reference table",
  "filterId": "0",
  "changeStatus": "updated",
  "graphics": [],
  "filterText": "All",
  "code": "S1000DBIKE-AAA-D00-00-00-00AA-00PA-D",
  "language": "EN-US",
  "type": "html",
  "version": "004:00"
}

```

Figure 21 Pinpoint document structure

7.3.3.1 WEBVIEW INTERFACE

To enable communication between the app and the WebView, our solution was using a 3rd party plugin available on GitHub. This lets us communicate between the WebView and DocumentView. This works by calling JavaScript functions through the WebView url.

Example of JavaScript url call `javascript:alert('alert function called')`

7.3.4 PINPOINT DOCUMENT VIEW

7.3.4.1 ROUTING

While routing is mostly handled by angular, it is not as easy to send an arbitrary number of variables to a route.

The routes for the document view

```
{ path: "document/:publication/:path/:filterIds", component: DocumentViewComponent }
{ path: "document/:publication/:path", component: DocumentViewComponent }
```

Figure 22 Document view route

The path requires a name and an optional filter id list with objects separated joined by '+'. The name is the document pointer as found in the pinpoint ToC file

More details on this can be found in Toc View

7.3.4.2 LOADING A DOCUMENT

Loading the HTML document gave us some challenges. The scripts must be loaded in the correct order, and the DocumentView component must be notified when the body is loaded so the applicability filter can be applied.

- es6-promise and jQuery are dependencies of other files, and should be included first
- nativescript-webview-interface is used by other files, and requires es6-promise to be loaded.
- openLink.js contains the implementation of the openLink function used throughout the pinpoint document. It should be loaded in the header after webview-interface
- applicability.js uses webview-interface modifies the body, and should be included at the end of the document

7.3.4.3 FILTERING ELEMENTS

To filter the documents based on applicability, we must find every element with an applic id attribute. This becomes a lot easier to manage through the jQuery library. Through jQuery we can find all elements with an applic id tag, find the element type and determine if it should be displayed or not

```
$('#[applicId]').each(function () {  
  
    var toFilter = $(this);  
  
    //If element is a div, filter the parent instead  
    if ($(this).is('div')) {  
        toFilter = $(this).parent();  
    }  
  
    //Hide element if it isn't applicable  
    if (isApplicable(this, filterIdList)) {  
        toFilter.css('opacity', '1');  
        toFilter.css('color', 'green');  
    }  
    else {  
        toFilter.css('opacity', '0.2');  
        toFilter.css('color', 'red');  
    }  
});
```

Figure 23 jQuery for filtering documentation

This function gets called when the page is loaded or when the applicability filter has changed.

The applicability filter is sent to the function as a parameter through the WebView interface.

7.3.4.4 HANDLING LINKS

The bike documents we received from Flatirons Solutions contains links to other documents and to images. While implementing these are not high on our priority, we have considered how to do it.

Since the WebView component from NativeScript accepts any valid HTML and behaves like a browser, we can link a JavaScript document containing the missing functions to implement the links. These must return information about what was requested to the NativeScript application. You can read more about the information transfer under WebView Interface.

7.3.4.4.1 LINK TO ANOTHER DOCUMENT

After clicking a link, the application will be routed to the same component. This is a bit problematic since angular doesn't reload the component if you route to the same one. The solution to this was to subscribe to the parameters in the route, and perform the desired action when a change is detected.

7.3.5 HTML DOCUMENT FACTORY

To prepare the pinpoint documents we created a HTML document factory. This allows us to define a template with common scripts and stylesheets, and easily add additional content. This is done by saving the input in arrays for header, footer and body elements. When the document is built, the elements are arranged in order

Structure of the generated document

```
<html>
  <head>
    HEADER ELEMENTS IN ORDER
  </head>
  <body>
    BODY ELEMENTS IN ORDER FOOTER ELEMENTS IN ORDER
  </body>
</html>
```

7.3.5.1 TEMPLATE

Since all the pinpoint documents uses the same styles and JavaScript files, we generated a template with pre-added headers and footers.

Template constructor

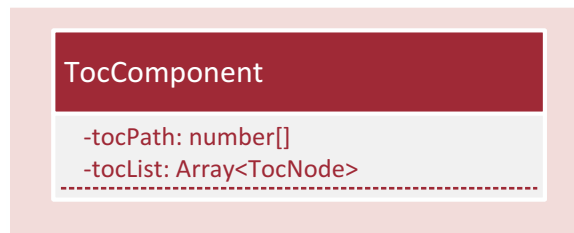
```
static get template (): HtmlDocumentFactory {
  /*
   * The applicability.js file has to be added after
   * declaring the body of the document.
   * Applicability.js relies on nativescript-webview-interface * which uses es6-
   promise.min.js,
   * so they have to be added first.
   * The openLink.js has to be added before the body.
   */
  return new HtmlDocumentFactory()
    .addStylesheet("./data/S1000D_Bike_Set/css/S1000DIssue4_0_main.css").addScriptHeader("./lib/jquery-3.1.1.min.js")
    .addScriptHeader("./lib/es6-promise.min.js")
    .addScriptHeader("./lib/nativescript-webview-interface.js")
    .addScriptHeader("./javascript/openLink.js")
    .addScriptFooter("./javascript/applicability.js");
}
```

```
<html>
  <head>
    <link rel="stylesheet" type="text/css"
          href="./data/S1000D_Bike_Set/css/S1000DIssue4_0_main.css">
    <script src="./lib/jquery-3.1.1.min.js"></script>
    <script src="./lib/es6-promise.min.js"></script>
    <script src="./lib/nativescript-webview-interface.js"></script>
    <script src="./javascript/openLink.js"></script>
    HEADER_ELEMENTS_IN_ORDER
  </head>
  <body>
    BODY_ELEMENTS_IN_ORDER

    <script src="./javascript/applicability.js"></script>
    FOOTER_ELEMENTS_IN_ORDER
  </body>
</html>
```

7.4 TOC VIEW

The toc view is the component that lets the user navigate the table of contents.



7.4.1 LOADING TABLE OF CONTENTS

7.4.2 TOC NODES

The ToC file that comes with the Pinpoint data set consists of nested ToC arrays. The root ToC object is a little different than the rest of the objects. We made two interfaces that describe the root object and the ToC nodes.

ToC Interface Objects

```
export interface TocNode {
  title: string;
  pointer: string;
  leaf: boolean;
  type: string;
  toc: TocNode[];
}
export interface RootObject {
  revisionNumber: string;
  toc: TocNode[];
  library: string;
  publication: string;
}
```

Example of nested ToC nodes

```
{
  "revisionNumber": "002",
  "toc": [
    {
      "title": "Bike 4.0.1 - Top",
      "toc": [
        {
          "title": "URLs, Hotspots and TIR Bike",
          "toc": [
            {
              "title": "Bicycle - URLs, Hotspots and TIR",
              "pointer": "DMC-S1000DBIKE-AAA-D00-00-00-00AA-041A-A_000_EN-US",
              "leaf": true,
              "type": "html"
            }
          ],
          "leaf": false,
          "type": "html"
        }
      ],
      "pointer": "PMC-S1000DBIKE-N4701-00001-01_001-01_SX-US",
      "leaf": false
    }
  ],
  "library": "4ade5e74-f9dc-4066-a100-23d6d849cfd3",
  "publication": "f9c66907-68af-4b45-8d72-50575cdf1aea"
}
```

7.4.3 NAVIGATING THE TABLE OF CONTENTS

When the user navigates the table of contents view move further into the tree for each click. We do this by generating an address for where the user is.

```
{ path: "toc/:publication/:path", component: TocComponent },
```

An example address for the example above would be: toc/:publication/0+0+0. This would give us the Bicycle- URLs, Hotspots and TIR node.

For each level the user goes through we route back to the table of content view with a path parameter. This lets us know what should be displayed. When the user reaches a node where the leaf parameter is true. We redirect the user to the document view using the node pointer.

7.5 BEACON COMPONENT

In this section, we will give a detailed explanation regarding the beacon part of the architecture. The first part introduces how we integrated the Estimote SDK. The second and third part explains how we range beacons for iOS and Android. The fourth part explains how we prepare the data we get from the beacons. The fifth part explains how we use the former sections to get list picker view.

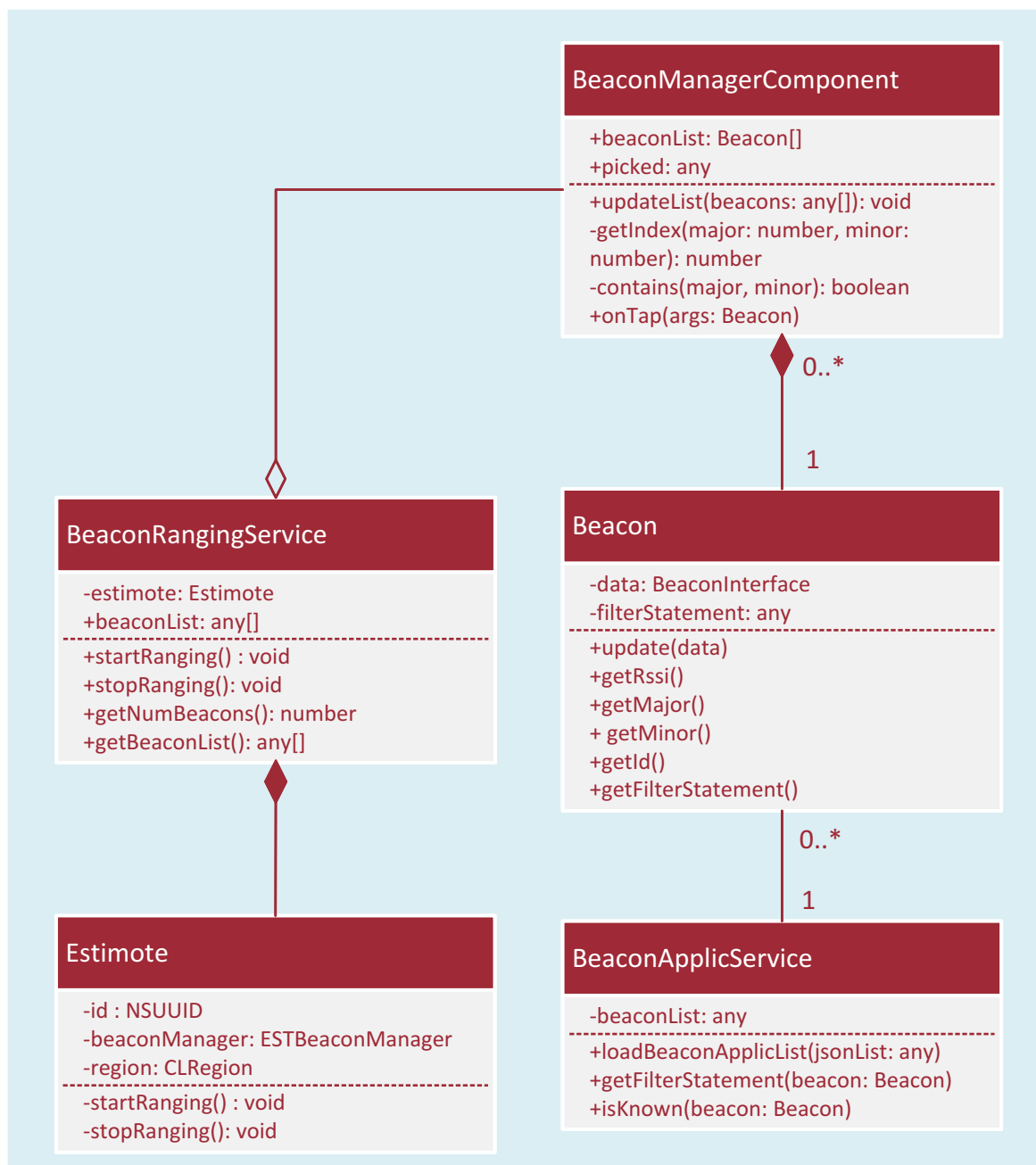


Figure 24 - Class diagram for beacon view

7.5.1 ESTIMOTE PLUGIN

Estimote has their own SDK provided for both Android and iOS. The SDK uses the native libraries for both platforms. This means that without a cross platform wrapper we would need develop two separate apps using the platforms own tools. Rather than doing that, we are using NativeScript which is a cross platform framework that wraps around the native libraries. To add the Estimote SDK to our NativeScript framework we needed to develop a plugin that integrates the android and iOS Estimote SDK.



🍏 iOS SDK and 🤖 Android SDK

Figure 25 Estimote SDK [28]

7.5.1.1 HOW TO ADD ESTIMOTE SDK TO OUR PLUGIN

There are several ways to add native SDK to a NativeScript plugin. The recommended way is to use Cocoapods for iOS and Androids dependency system. This way we ensure we can get the latest SDK. [29]

ANDROID

Adding Estimote SDK to android was straightforward. In plugin/platforms/android we make a file called include.gradle. Here Estimote's package address should be included under dependencies. For permissions, we need to add a AndroidManifest.xml file to specify which permissions we need.

IOS

Adding Estimote SDK to iOS was not straightforward. This was something we had a lot of challenges with and took some time to resolve. We found a good solution. For iOS, we create a Podfile under plugin/platforms/iOS. This Podfile includes the name of the Estimote Cocoapod name. We are now able to get the EstimoteSDK from Cocoapod and it is fetched when we build. The problem was that NativeScript was not able to generate any metadata.

We figured out that the EstimoteSDK was missing a module.modulemap file that describes the framework for the compiler. So a temporary fix for this was to go inside the build and edit in a Modules folder that contains a module.modulemap file that describes how to get the metadata from the Estimote SDK headers. Right now, the plugin can't auto generate this folder inside the Estimote framework folder. When this is done the compiler now finds the framework and we can build our application. [30]

AUTO COMPLETION

To more easily work with the EstimoteSDK we want to be able to get auto completion for our text editor. This is done by generating typings for the EstimoteSDK and then using references to these typing's. A temporary solve that we have now is we generate the typing's, move them to the platform declarations and then include them in the specific reference index file for that specific platform.

7.5.2 BEACON DETECTION FOR IOS

We have created a Estimate class that we can use to start the beacon ranging. We also have a BeaconManagerDelegate which gets the updates for the beaconManager.

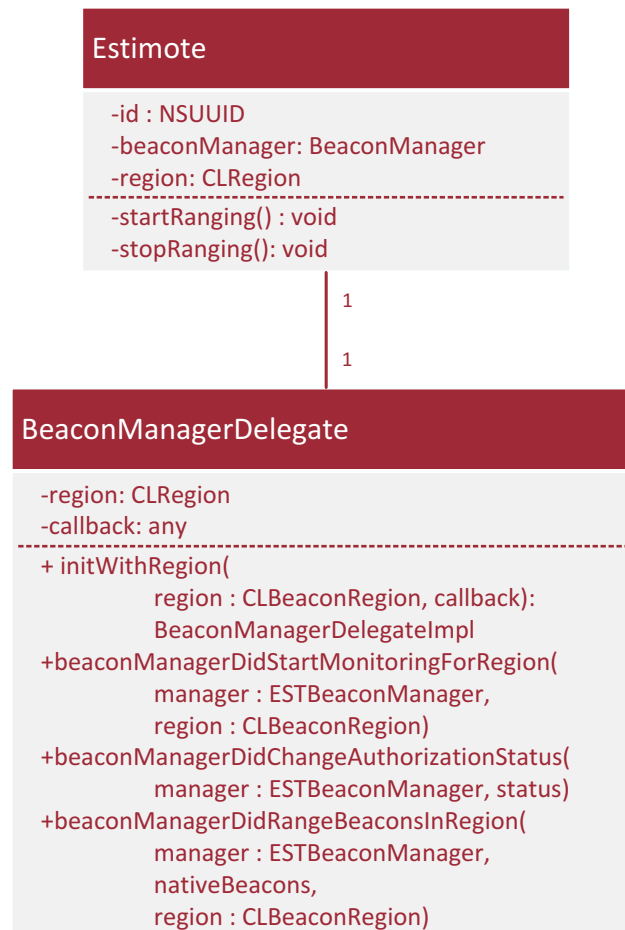


Figure 26 - iOS class diagram for ranging beacons

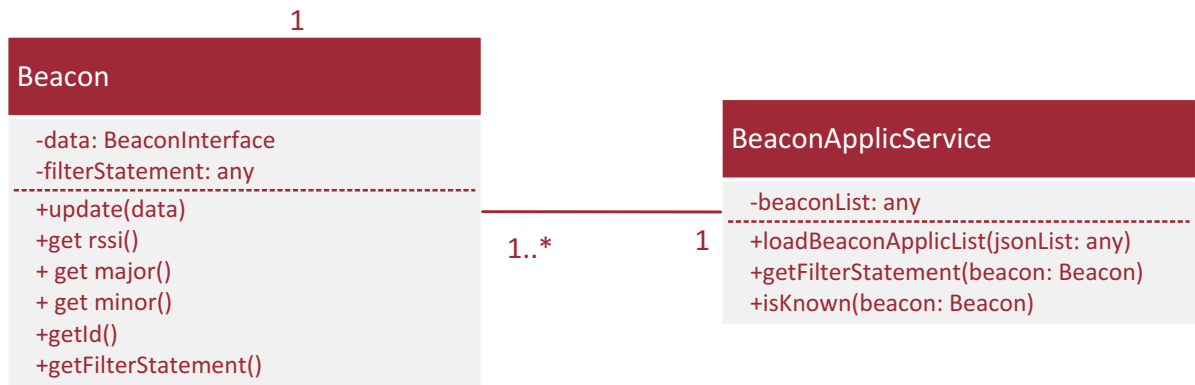
When we initialize a Estimate object we set beaconManagers delegate to BeaconManagerDelegate. This lets us get updates with new beacons that are detected in the region specified. The nativeBeacons is a list of the nearby beacons. We then pass the list to the callback.

7.5.3 BEACON DETECTION FOR ANDROID

We use a similar Estimate class as we do with iOS for android (Figure 26). The functions are all the same and we use a call-back to get the beacons in range. When we initialize a Estimate object we set a range listener that listens for nearby beacons. When the listener finds a beacon, a function gets called and we can pass the list of beacons using the call-back. The startRanging function checks for all required permissions have been given before we start ranging.

7.5.4 PREPARING BEACON DATA

We have a beacon model that represents each beacon. For each Beacon object we create, we also give it a filterStatement from the BeaconApplicService. The BeaconApplicService uses a JSON file to cross reference beacons.



7.5.4.1 BEACON CROSS REFERENCING

We are using the Estimote beacons to filter the applicability for our application. To do this we have created a JSON file that contains information regarding each beacon. Since the beacons transmit their minor and major value we are using this to tell them apart. An example looks like this:

```
"1675-12907": {  
  "name": "Candy Floss",  
  "display_text": "Mountain Storm mk1",  
  "publication": "f9c66907-68af-4b45-8d72-50575cdf1aea",  
  "filterStatement": {  
    "type": "Mountain bicycle",  
    "model": "Mountain storm",  
    "version": "Mk1"  
  }  
}
```

Where 1677-12907 is the major and minor and everything inside the curly brackets are information about this exact beacon.

- Name: what the beacon is called
- Display_text: what the beacon is displayed as in the beacon list
- Publication: which publication the beacon is relevant for
- Filterstatement: The filter statement for the beacon.

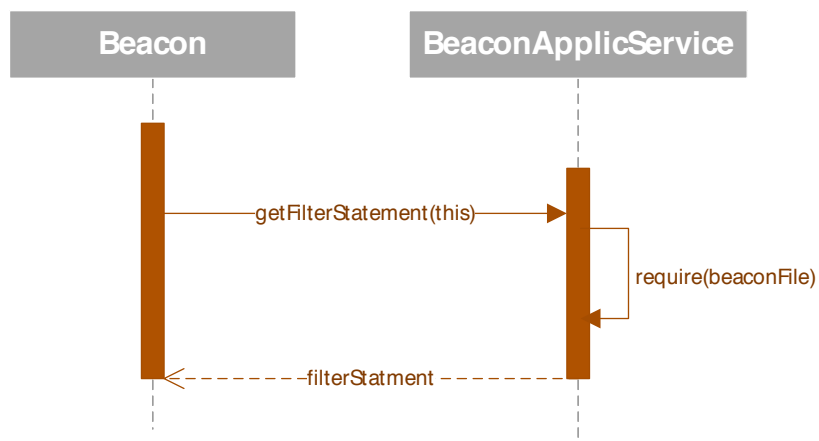


Figure 27 - Getting filter statement

7.5.5 BEACON VIEW

The beacon view contains the UI of the beacon part of the application. We have a list of beacon objects that gets displayed in a list picker. When the view gets created we start a timer that requests a list of nearby beacons from the BeaconRangingService every few seconds. The BeaconRangingService starts the Estimote ranging and updates a list of beacons for each discovery. We then update the list picker with the new beacons. The list is sorted by using the RSSI value of each beacon to determine the distance. When the user submits a beacon the view returns the selected beacon object. We use that object to filter the documents.

7.5.5.1 UPDATING BEACON LIST VIEW

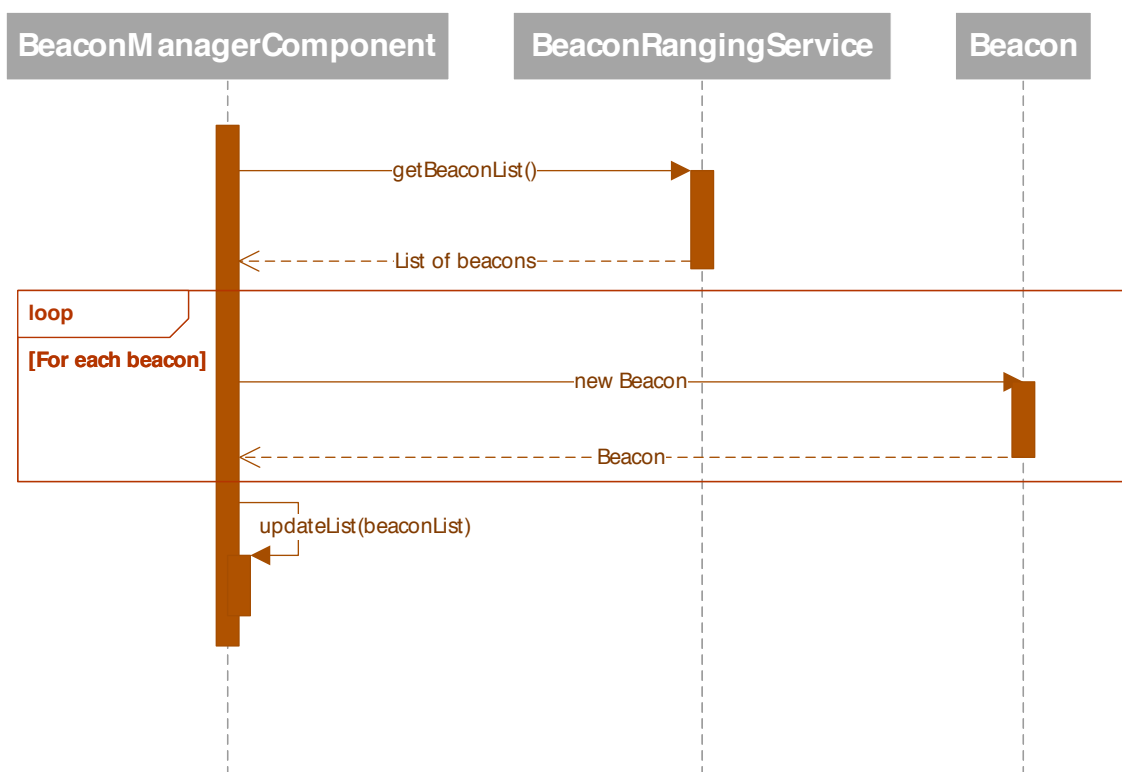


Figure 28 Display beacon list

7.6 FOLDER STRUCTURE

The main contents of the application are inside the app folder.

- Applicability
 - Consists of the applic filter part of the software architecture.
- Beacon
 - Consists of the beacons part of the software architecture.
- Document-manager
 - Consists of the publication manager part of the software architecture.
- Document-view
 - Consist of the Document viewer part of the software architecture.
- Toc
 - Consists of the Toc part of the software architectur
- Lib
 - Consists of different JavaScript libraries we have used for the application.
- Tests
 - Consists of the unit tests we have written for the application
- JavaScript
 - Consists of JavaScript files that gets included in the HTML factory.
- Data
 - This folder holds the publications. DC9 and S1000D_Bike_Set
- App.component is the main view component.
- App.routing keeps track of routing to different components
- Beacon.JSON keeps a cross reference of beacons and their filterstatement
- App.module is the main module of the application.
- App_resources
 - Consists of platform specific resources such as permission and assets.

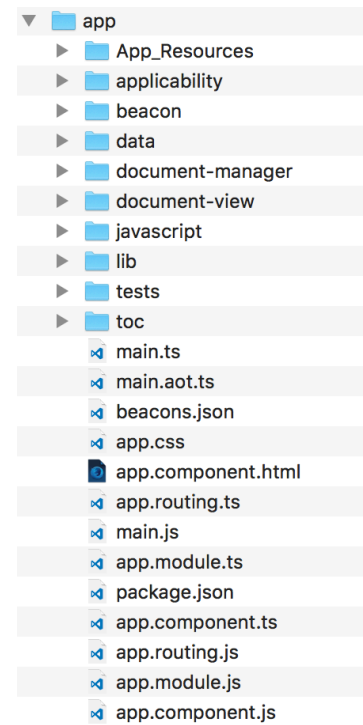


Figure 29 - Folder Structure

8 EVALUATION

In this chapter, we discuss the requirements, challenges we met and a practical evaluation of the project. We have also included the feedback from a demonstration with a former flight technician.

8.1 HAVE WE MET THE REQUIREMENTS?

When we look at the requirements in chapter 5 we can see that Flatirons Solutions have set 5 different requirements. These are:

- Cross platform
- Use beacons to filter applicability
- Use a mock data set

Requirements added during the project:

- Use DC9 data set
- The ability to swipe through documents

When we finished the development phase of our project our application accept data sets that are in the format Pinpoint neutral package. This means that both the use of a mock data set and use of a DC9 data set requirements have been met. The application also lets the user filter the content of those data sets using beacons and it works for both iOS and android with some minor differences. Taking this into account we can fairly say that the application meets the requirements that Flatirons Solutions have provided during the project.

8.2 CHALLENGES

Through the project, we have faced several challenges. In this section, we will give you a insight into those challenges.

8.2.1 INTEGRATING BEACONS IN NATIVESCRIPT

NativeScript does not have official support for implementing beacons. When we picked NativeScript we knew this and did account for the beacons part to take some extra time. We knew NativeScript supported native code and that it seemed trivial to include a 3rd party SDK. When it came time to integrate the Estimote SDK into our NativeScript application we found out that it didn't work as planned. The steps in the NativeScript guide for implementing 3rd party SDK's didn't not work on iOS. For android, it worked as planned. We did not have enough knowledge regarding NativeScript plugins or how we would integrate a 3rd party SDK for iOS. After several days of not making progress we figured it out and you can read about how we managed to integrate the SDK for iOS in chapter 5.4.2.

8.2.2 IMPLEMENTING SWIPE ON ANDROID

When we implemented the swipe functionality everything worked fine for iOS. When we tried the same implementation on android swipe did not work. After some research, we figured out that the android web view's zoom controller consumes all touch gestures. What we did to make swipe work was to disable the web views zoom controller.

8.2.3 NEW DEVELOPMENT TOOLS

None of us had ever developed a mobile application or used beacons. Everything we were going to use was new. This was a challenge since we would need to learn TypeScript, NativeScript, Angular 2, QUnit for testing and Git for source control.

8.2.4 NO ONE HAS DONE IT BEFORE

When we first started with this project we tried to find similar projects that had been done before. We found two master theses from the university of Oslo. One that used beacons to enhance the experience at a museum and the other to register tickets for a metro system. We have not found any papers or projects regarding the use beacons to filter technical documentation. This is something no one has done before us.

8.2.5 DC9 CHALLENGE WITH IOS

Loading the table of contents for the dc9 data set on iOS creates challenges for the beacon ranging. When we load a large document, the beacon ranging callback is set to undefined. If we use a smaller table of content, this does not happen. This is a bug in the framework for iOS and we have chosen to implement a workaround.

8.3 PRACTICAL EVALUATION

8.3.1 ESTIMATE BEACONS

During this project, we mostly had only positive experiences using the Estimote SDK and Estimote beacons. We had some difficulties implementing the Estimote SDK into our NativeScript project but that was not Estimote's fault. The beacons have been reliable throughout the 5 months we've worked with them. One of the great features Estimote have is their app to program the beacons. We can easily change any options the beacons let us, with their mobile application.

They have good API documentation and examples. The SDK is still in early stages and the android version had a 1.0.0 release candidate update in the middle of our project. This caused some changes that was breaking, but there was not a problem fixing this.

8.3.2 USAGE OF NATIVESCRIPT

We started this project without any previous knowledge of NativeScript. This was a framework Flatirons Solutions recommended we use. During our project, we have gained some experiences using NativeScript.

One of the good elements with NativeScript is the fact that you can quickly prototype for both iOS and android. We have made a list of pros and cons we have based on our experiences.

- Positive
 - Familiarity with programming language
 - Quick prototyping
 - Has TypeScript typing's for NativeScript API.
 - Access to entire Native API
 - Angular 2 support Collaboration with Google's angular 2 team.
 - CSS support
- Negative
 - Poorly explained basis from tutorial
 - Platform inconsistencies
 - Obscure error logs
 - Don't feel we have full control over native side
 - Still under development. Unstable API
 - Small community

8.3.3 BEACON ACCURACY

A question we wanted to answer was "how accurate is the distance between beacons?". We performed field test to record the RSSI of the beacons at different distances. We measured the signal strengths using both iOS (iPhone 7) and Android (Galaxy S7) devices from these lengths:

- 1m
- 1.5m
- 3m
- 6m

To analyse and filter the signals, we have traded knowledge with another student group, Stochastical Engineering. With their assistance, we have been able to plot, and apply a simple low-pass filter to the raw signal. A low-pass filter is the moving average of the previous N samples, where N is the filter order. Each beacon has their own name Candy, Lemon and Beetroot. For all the figures the distances for each sample is listed here:

The Different distances for ios1 and android1.

- Candy is 1 meter away from the phone.
- Lemon is 1.5 meter away from the phone.
- Beetroot is 3 meter away from the phone.

The different distances for ios2 and android2.

- Candy 1 meter away from the phone
- Lemon 3 meter away from the phone
- Beetroot 6 meter away from the phone

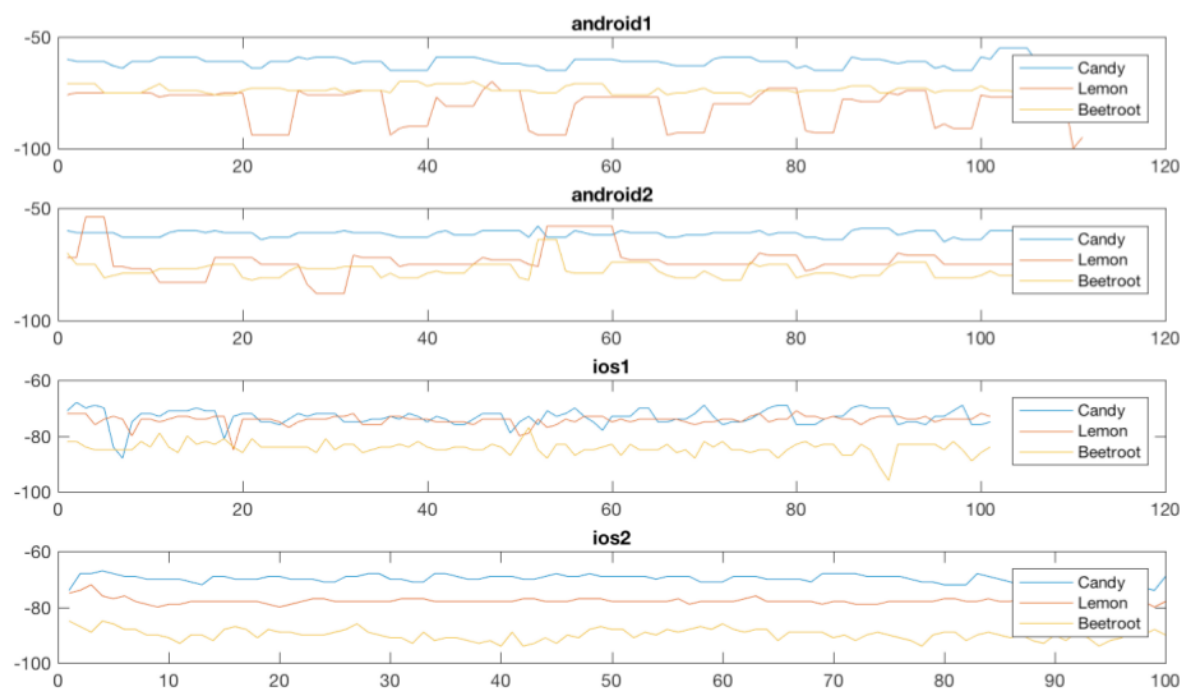
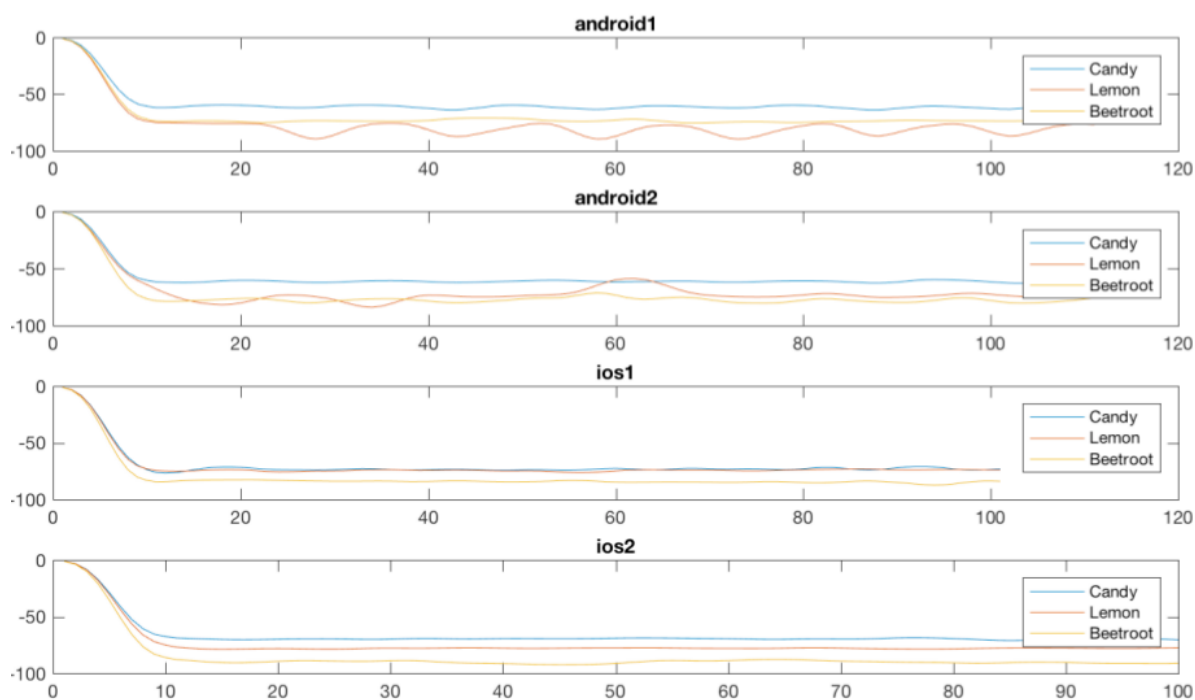
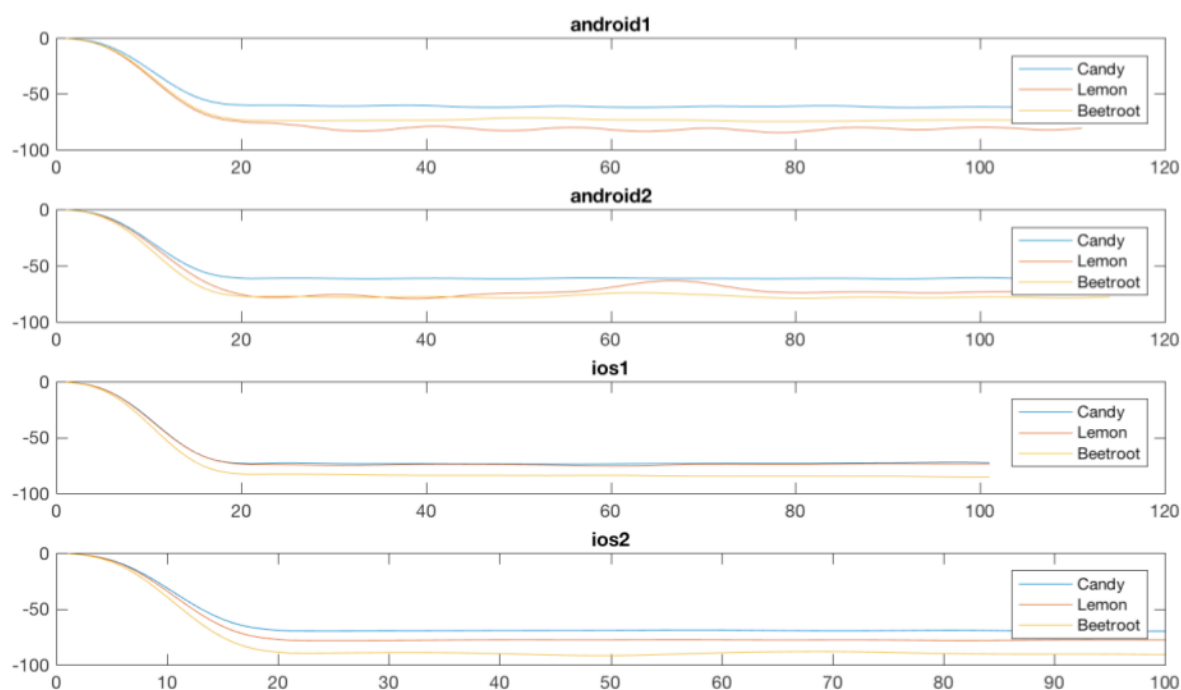


Figure 30 - unfiltered signal

Figure 31 – 10th order low pass filterFigure 32 – 20th order low pass filter

With 20th order low pass filter, iOS is very stable and gives the values we would expect. Android still detects Beetroot as closer than Lemon, and would give us the wrong order if we ordered the beacons from closest to furthest away.

To get a good filtering, we would recommend using at least a 10th order low pass filter. The recommended broadcasting interval is 100ms for iOS and would take 2 seconds to get a good sample. This should not be a problem since we don't want to update the range of the beacons every too often. By updating the beacons more frequently with fewer packets, the measured RSSI would become unstable.

8.4 DEMONSTRATION

We held a demonstration of the mobile application to an employee at Flatirons Solutions the third of May 2017 at Flatirons Solutions Kongsberg premises. The employee was a former airline technician and familiar with the maintenance routines regarding airplanes. We thought it would be beneficial to collect a professional opinion from a technician on the application we had developed. We showed him the mobile application and the beacons to see if the concept of using beacons to filter content in a data set would be useful to a technician. Although he is quite at age, he quickly and intuitively managed to use the mobile application and understood the purpose of the system.

| Students: | Employee: |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | <p>"This was stylish".</p> <p>"So this will give me the relevant documents where I am located at the airplane?"</p> |
| <p>"We can make that happen, but now the beacon only gives you the applicable documents for that particular airplane".</p> <p>"Do you think this (mobile application) is relevant for a technician?"</p> | |
| | <p>"Yes, I do think so. It may help at fault prevention. (Meaning technician gets the right documents)."</p> |
| <p>"Would it (mobile application) make your maintenance job any easier?"</p> | |
| | <p>"Yes, it is more economical and secure way of doing it instead of back in my time, where we had instructions printed out on paper. And I like that it is held updated, in case of changes in the documentation"</p> |
| <p>"Do you have any ideas to improvements or anything you want to change in the mobile application?"</p> | |
| | <p>"No, not at the moment. I like the big buttons and the interface. Remember that a technician may be out in the cold and have big brute fingers."</p> |

He then proceeded to give us a recommendation to continue developing the system and make sure Flatirons Solutions got a demonstration of the mobile application.

9 CONCLUSION

In this bachelor project the team working on the solution invented an unprecedented innovative product. This product serves the engineering trait to its core. It is the pure essence of enhanced forward thinking holistic approach to engineer a life cycle solution to a marked demand.

The product developed called BApp and the massive amount of research accumulated to produce a working product will see its future in all technical fields. BApp is a generic solution which can be used in other fields. Ranging from big industries like the airline and aircraft industry and defence and aerospace to everyday common use such as in household appliances, in a museum or in public transport systems. We have created BApp with CORENA Pinpoint in mind. The solution presented in this project is tailored to fit Flatirons Solutions requirements, but it works for any other document you can imagine. The team have created a component based application, where the user can add a database and add beacons or other elements, to upload pdfs or links to user manuals to get for example a smart house.

9.1 CONTRIBUTION

9.1.1 FUNCTIONAL FOR FLATIRONS SOLUTION

One of the functional requirements is the bachelor group achievement is to produce a swipe function for Flatirons Solutions. This is a feature Flatirons Solutions do not have, and this will be a great advantage for the company.

9.1.2 PRECEDENCE

To use beacons to filter documents in a data set have never been done before. This will give Flatirons Solutions an innovative advantage within safety and economy. The team is proud to present BApp, full functional mobile application to fulfil the requirements from Flatirons Solutions.

9.1.3 KNOWLEDGE

An application like this has never been created before. A huge amount of research has been done to gain adequate knowledge to produce a solution, and to take the project further. In this generic solution, we have designed the application to support different types of documentation by only replacing a few components.

9.1.4 FUTURE WORK

We have gained and documented new technology. We have considered other hardware and other solutions, such as AR and VR concepts.

9.1.5 NOVELTY

The whole project has a novel stamp on it. All is new and original. BApp uses beacons to filter documents in a data set – this is a novel approach to document filtering.

9.1.6 DOCUMENTATION MANAGEMENT

Revolutionary use of new hardware to use existing software better and tweaking existing solutions to perform more efficient.

9.1.7 SAFETY AND ECONOMY

BApp reduces human error and decreases man hours by automatic presenting the correct data set. This will save time and money and user errors. This will give Flatirons Solutions a competitive advantage in the marked.

9.2 RELATED WORK

We have found two similar projects:

USING BLUETOOTH BEACONS TO PAY FOR METRO TICKETS

A case study in user control of ubiquitous computing. An Android application was created and tested on users. The participants felt at unease when the system automatically purchased a ticket. Users felt that some control still desired. The thesis concludes that a transition to an environment of ubiquitous computing systems should be just that: a transition.

<https://www.duo.uio.no/bitstream/handle/10852/51097/1/Torkjel-Skjetne-Masteroppgave-mai-2016.pdf>

USING BLUETOOTH BEACONS IN A MUSEUM: AN EXPLORATORY STUDY WITH PROXIMITY-BASED TECHNOLOGY

The goal of this master thesis is to discover how Bluetooth beacons affect user experience in a museum setting, specifically for museum visitors and staff. This thesis also concludes that some form of user control is still desirable.

<https://www.duo.uio.no/handle/10852/51251?show=full>

The two similar projects both explore how the user perceives a fully automatic system. In this project, we have focused on the implementation and integration of beacons in technical manuals.

The application that has been developed in the project has a component based architecture. This makes it possible to act as a generic solution and can be used for more than just technical manuals.

9.3 FUTURE WORK

During the project, we have discussed several features we would like to try out. We estimated it was no time to develop any of these features. In this section, we'll give you a quick overview of the features we think should be looked further into.

9.3.1 BEACON LOCATION

We have discussed several possibilities on how to use beacons. A very interesting idea is to use beacons to determine the user's location.

We had a meeting with a Flatirons Solutions discussing possible solutions for beacons. An interesting idea was using beacons to pinpoint specific parts of the airplane. An example for instance could be the user is standing close to the middle of the plane. Could we then determine the user's location and filter information based on his location?

We have some ideas on how it would be possible to solve this. We have three different suggestions:

- Triangulation
- Estimote Indoor Location SDK
- Multiple beacons with short range

9.3.1.1 TRIANGULATION

We can use the RSSI to calculate the distance between two or more devices.

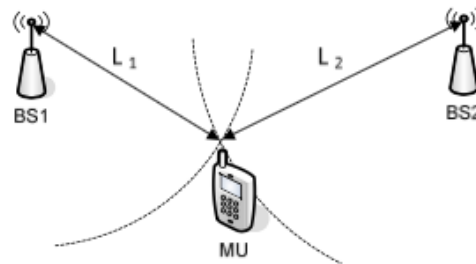


Figure 33 - Measuring method [25]

But since this method is device dependent, it is unreliable and there for needs a special method to get more usable RSSI value. Triangulation is one of these algorithms and it calculates the location based on distances between beacons and the mobile. This method forms circles centred at the beacons, where the radius of each circle is determined by the measured signal strength of the mobile. An intersection points arises when there are three or more beacons within a certain range, then the intersection point gives the estimated location of the mobile. Because of signal can be affected by obstacles, the final position estimation will be the average coordinate of intersection points. (Figure 34)

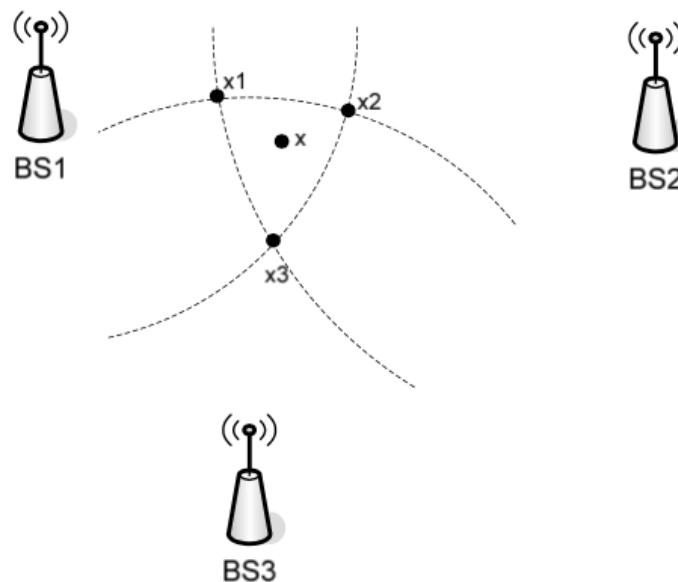


Figure 34 - Triangulation with multiple intersection points [25]

As there could be many obstacles between beacons and mobile and diffraction by these obstacles cannot be ignored, therefore an adopted sophisticated indoor propagation model could be useful.

$$RSSI = (10n \log_{10} d + A)$$

The Radio Frequency (RF) parameters are used to describe the network environment. the RF parameter A is defined as the absolute energy which is represent by dBm at a distance of 1 meter from the transmitter, which is RSSI reading at 1m from the transmitter; n is the signal transmission constant, and it is relevant to signal transmission environment; d is the distance from the transmitter node to the receiver node.

When the mobile does not enter the positioning area, there are only the reference nodes in the positioning area. Therefore the environmental factor between two reference nodes can be estimated as

$$n_i = \left(\frac{RSSI - A}{10 \log_{10} d_i} \right)$$

Figure 35 - RSSI distance formula

CENTROID POSITIONING

The geometric centre of a polygon is called the centroid. The average value of polygon vertex is the coordinates of the centroid node. For the Centroid positioning algorithm we need firstly determine the region that contains the mobile, and then calculate the centroid of the region. The centroid acts as the location of mobile. For example in *Figure 36*, the measured RSSI values from the mobile to reference nodes can yield four distances. We can draw four arcs using these distances from the corresponding reference nodes and we can get four intersecting point A, B, C and D.

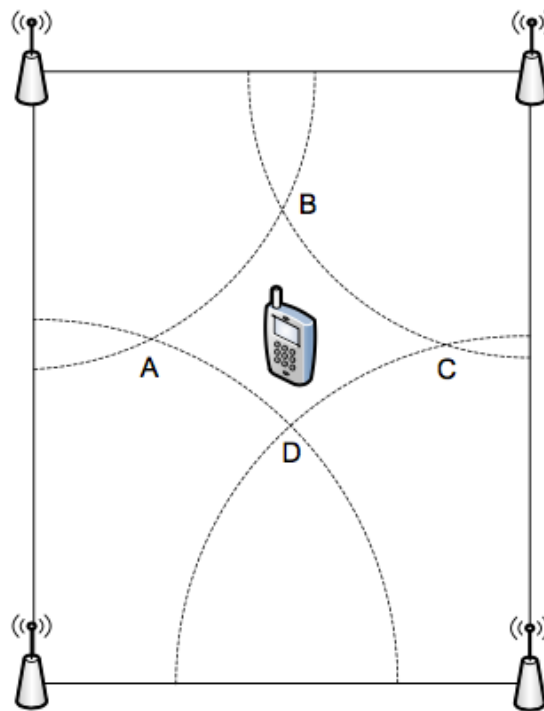


Figure 36 - Centroid positioning concept [25]

Set the polygon defined by A, B, C and D with coordinates $A(x_1, y_1), B(x_2, y_2), C(x_3, y_3), D(x_4, y_4)$ the coordinates of the mobile (x, y) is the polygon's centroid:

$$(x, y) = \left(\frac{x_1 + x_2 + x_3 + x_4}{4}, \frac{y_1 + y_2 + y_3 + y_4}{4} \right)$$

Figure 37 - Location from distance radius intersections

As conclusion, it is worth to say that RSS based triangulation positioning yields very good results.

[31] [32]

9.3.1.2 ESTIMOTE INDOOR LOCATION SDK

Estimote has an Indoor location SDK where it is possible to determine the users position down to a 1 meter accuracy in small rooms and 4 meters in large rooms. Using 4+ beacons you can calibrate the space and pinpoint the user to his location. This could be useful inside an airplane to know exactly where the user is located. The upside to this solution is the need of beacons. We believe we would need less beacons to pinpoint the user accurately than using multiple beacons with short range. The downside is the fact that you need to be inside the grid of the beacons making filtering of information outside the plane difficult.

9.3.1.3 MULTIPLE BEACONS WITH SHORT RANGE

Using short range beacons, we can create small beacon zones inside and around the plane. An example could be a beacon that represents the engine. When the user enters a zone, we would filter the information relevant to that zone. Using a ID for the model and a ID for the zone. As of today, the data set we have access to does not have the possibility of applying applicability to different zones, but this is something we believe could make the search for the right information easier for the user and should be researched further for anyone wanting to pick up on this work.

9.3.2 IMPLEMENTATION

9.3.2.1 ASSERT RANGE + MULTIPLE VALUES FIX (APPLICABILITY)

`applicPropertyValues= "2 | 5"`

To represent a range of values, separate the values with a tilde "~"

`applicPropertyValues = "5 ~ 7"`

These can be mixed to select multiple ranges or values

`applicPropertyValues = "2 | 5 ~ 7 | 9 ~13"`

The result of the applic evaluate must be true for the settings is applicable

The <Assert> element tells you what values of the identifier something is applicable. An assert can be used to represent different types, and ranges. To represent multiple values, they may be separated by a vertical bar "|".

The prototype we have developed do not account for different ranges. To do this we need to evaluate the ranges. The different value ranges in `applicPropertyValues` needs to be split. The value ranges then need to be split to find the min and max range.

9.3.2.2 BUFFER NEXT / PREVIOUS DOC (SWIPE)

When the user swipes a document, the user will not get to see the next document before it has been routed to. We think the user experience of the swipe could be improved by showing the next document as an animation. To make this work we need to buffer the next and previous document.

9.3.2.3 PROPER IMPLEMENTATION OF SWIPE (TRACK FINGER MOVEMENT)

The swipe function as it is implemented now is far from perfect. It relies on events registered by the components, but this functionality is not identical on iOS and Android

The swipe function we have implemented is just for a prototype. A proper way to implement the swipe function would be to track the finger movement. This would allow us to swipe documents even when we are zoomed in the document. Tracking the finger speed and direction we could guess if the user wanted to move around in the document or if the user want to change to another document.

9.3.2.4 HIDE NOTIFICATION AFTER VIEWING

The “beacons” button changes colour when any registered beacons are nearby. This means that the user is always notified when beacons are available, but not when a new beacon is detected.

To improve this, the notification colour should be removed after the user has viewed the beacon list, and the user should be notified again when a new beacon is available.

9.3.2.5 SHOW WHICH FILTER IS APPLIED

There is no indication anywhere in the UI which applicability has been selected. To know what filter is applied the user must remember which beacon was selected, or reapply the filter. This does not lead to a good user experience, and the information should be available somewhere in the application.

9.3.3 COMPONENT IDENTIFICATION

The team considered how we could identify components. We think the beacons are not suitable to identify a single component. Instead we think beacons should be used for the identification of the model and using any of these technologies for each component identification:

QR

Quick Response Code: It is a 2D matrix Bar Code label. The information is stored both horizontally and vertically. A single QR symbol can store up to 7,089 digits contains numbers and characters

NFC

Near Field Communication and is a set of short distance wireless communication solutions based on RFID and internet technology. It is a communication protocol that enables two NFC devices to communicate. The NFC tags can be passive meaning they will not need any power to function. It will get power from the mobile application when it is in range, which is within 10 cm from the NFC tag. [33]

RFID

Radio Frequency Identification(RFID) is a small chip and even smaller antenna. It is a considered a passive device thus not requiring any power. We can use a reader that can identify the RFID tag.

All three have their strength and weaknesses. The QR codes is easy to implement and only requires to be attached to the part that we want to be able to identify. The weakness is that the part needs to be visible. This is where NFC and RFID has their strength. NFC can be discovered up to 20 cm while RFID can be discovered up to a few meters. Using RFID tags using the ISM band we could get a range from 1m to 12m. More than enough to be able to list nearby components for the technician.

9.3.4 STORING FILTER ID ON BEACONS

When the user selects a beacon in the current application we use a cross reference of the beacons minor and major value assigned to the specific applicability. This requires that a publication contains such a file and is not standard for the pinpoint neutral package. A way to avoid having to cross reference using such a file would be to let the beacon broadcast the filter id. The Estimote beacons have their own custom protocol that could enable this. The beacons also have a 1mb of EEPROM storage which is more than enough to store the relevant information for the applicability.

9.4 CONSIDERATIONS

9.4.1 POWER CONSUMPTION

If you ask someone if they keep their Bluetooth on you often get the answer of no. It seems common conception of Bluetooth is that it consumes battery and increases power consumption a fair amount. [34]

9.4.2 MULTI THREADING

The application only uses one thread. The intention is that a server will contain the data set. Hence multi-threading in the application is not needed and not a requirement/priority in developing our application.

9.4.3 SECURITY

We only use the ID from the beacon to use in the data set. The data set is stored at a server. We have no sensitive data to secure. Someone could piggyback on our beacons, but this will not corrupt our data.

9.5 OTHER POSSIBLE SOLUTIONS

When we first started this project, the focus was to find out how we could use beacons to improve technical publications. The past few years more airlines are installing Wi-Fi to their aircrafts. We think it could be possible to use the routers service set identifier (SSID). The SSID is an identifier that is intended to be unique for a particular area. Instead of purchasing beacons and maintaining them we could use this identifier for each router to specify which airplane the technician is close to.

9.5.1 WIFI

We considered using the Wi-Fi from the plane itself. A problem here is that the aluminium fuselage of the airplane would block Wi-Fi signals. Wi-Fi most commonly uses the 2.4 gigahertz (12 cm) UHF and 5 gigahertz (6 cm) SHF ISM radio bands. [35] This solution could save costs when it comes to large scale deployment of beacons. Since most airlines are starting to roll out Wi-Fi on board their planes.

Positives

- Less costs if the airline already has Wi-Fi installed on their planes
- Less maintenance since we will not need to manage beacons batteries.

Negatives

- The user needs to be connected to the Wi-Fi
- Wi-Fi do not penetrate aluminium fuselage.

9.5.2 AUGMENTED REALITY

In the effort to create sustainable solutions to solve the aircraft and airline industries eternal quest to minimize human errors and reduce cost for the consumer, the team investigated the possibilities to achieve new and ground-breaking ways of using technology to meet tomorrow's demand for efficiency and security. Among these ideas the AR (Augmented Reality) emerged. This is the path the team recommend to pursue. This idea presents the feature that the user of the system is receiving relevant information on the very screen in front the user's eyes. Information of the desired component will be displayed directly on the screen to guide the technician to execute a safely and cost effective job.

Augmented Reality (AR) is a variation of Virtual Reality. [36]. It is a technology that superimposes a computer-generated image on a user's view of the real world, thus providing a composite view. [37] By using AR glasses the user will get relevant information on the lens. This could be an enhancement of the mobile application.

The bachelor members have also considered VR (Virtual Reality) for training purposes and machine learning that can be used under actual maintenance. Where machine learning also will give picture recognition for the AR system.

9.5.3 MACHINE LEARNING

Machine learning and neural networks can be used to train a computer in pattern recognition. This is applicable when looking at the location of a phone relative to beacons.

By training the application, it can learn and adapt to the patterns in the RSSI.

10 REFERENCES

- [1] Embraer, "ainonline," 05 May 2017. [Online]. Available: <http://www.ainonline.com/aviation-news/business-aviation/2015-06-28/oems-put-technology-work-mro-operations>.
- [2] Flatirons Solutions®, "Linkedin," 05 May 2017. [Online]. Available: <https://no.linkedin.com/company/flatirons-solutions>.
- [3] Angular, "Github - angular," [Online]. Available: <https://github.com/angular/angular>. [Accessed 10 April 2017].
- [4] Angular, 05 May 2017. [Online]. Available: <https://angular.io/>.
- [5] Google, "Routing & Navigation - ts - GUIDE," [Online]. Available: <https://angular.io/docs/ts/latest/guide/router.html>. [Accessed 11 May 2017].
- [6] J. Zhekov, "NativeScript Runtime Preview for Windows 10," 17 March 2016. [Online]. Available: <https://www.nativescript.org/blog/nativescript-runtime-preview-for-windows-10>. [Accessed 11 May 2017].
- [7] NativeScript, "Angular for NativeScript," 06 May 2016. [Online]. Available: <https://www.nativescript.org/nativescript-is-how-you-build-native-mobile-apps-with-angular>. [Accessed 16 May 2017].
- [8] Nativescript, 05 May 2017. [Online]. Available: <http://docs.nativescript.org/tutorial/chapter-0>.
- [9] "TypeScript," Microsoft, [Online]. Available: <http://www.typescriptlang.org/index.html>. [Accessed 25 April 2017].
- [10] B. S. Group, "Bluetooth Core Specification 5," Bluetooth SIG Group.
- [11] J. Xiong, Q. Qin and K. Zeng, "A Distance Measurement Wireless Localization Correction Algorithm Based on RSSI," in *2014 Seventh International Symposium on Computational Intelligence and Design*, Hangzhou, China, 2015.
- [12] E. Jackson, "Forbes," [Online]. Available: <https://www.forbes.com/sites/ericjackson/2014/11/29/interview-with-steve-cheney->

- on-estimote-and-the-power-of-local-commerce/3/#20fee165794c. [Accessed 28 April 2017].
- [13] Estimote, "twitter.com," 05 May 2017. [Online]. Available: <https://twitter.com/estimote>.
- [14] Wikipedia, "S1000D," 07 January 2017. [Online]. Available: <https://en.wikipedia.org/wiki/S1000D>. [Accessed 07 April 2017].
- [15] ASD, AIA, A4A, "S1000D - Home," [Online]. Available: <http://public.s1000d.org/Pages/Home.aspx>. [Accessed 07 April 2017].
- [16] "S1000D Issue 4.0," p. 1507.
- [17] S. Ambler, "Introduction to Test Driven Development," 19 April 2013. [Online]. Available: <http://agiledata.org/essays/tdd.html>. [Accessed 20 April 2017].
- [18] xarawn, "File: TDD Global Lifecycle.jpg," 04 Nov 2015. [Online]. Available: https://commons.wikimedia.org/wiki/File:TDD_Global_Lifecycle.png. [Accessed 18 May 2017].
- [19] K. Beck, in *Test-Driven Development by Example*, 2000, p. 15.
- [20] R. Osherove, "The Art of Unit Testing - Definition of a unit test," 2011. [Online]. Available: <http://artofunittesting.com/definition-of-a-unit-test/>. [Accessed 20 april 2017].
- [21] "Tutorialspoint.com," [Online]. Available: https://www.tutorialspoint.com/software_testing_dictionary/unit_testing.htm. [Accessed 09 May 2017].
- [22] DDI Development, "Scrum and Agile: when you should adopt them...," [Online]. Available: <http://ddi-dev.com/blog/it-news/scrum-and-agile-when-you-should-adopt-them/>. [Accessed 23 May 2017].
- [23] B. Roseke, "The Risk Management Process," 07 April 2016. [Online]. Available: <http://www.projectengineer.net/the-risk-management-process/>. [Accessed 08 April 2017].

-
- [24] T. Z. Warfel, Prototyping, Rosenfeld Media, 2009, pp. 1-5.
- [25] Office of Safety & Mission Assurance, "NASA Glenn's Facilities Division's System for Tracking Risks," [Online]. Available: <https://sma.nasa.gov/news/articles/newsitem/2015/08/13/nasa-glenns-facilities-divisions-system-for-tracking-risks>. [Accessed 23 May 2017].
- [26] Angular Team, "Angular Application Architecture," [Online]. Available: <https://angular.io/docs/ts/latest/guide/architecture.html>. [Accessed 18 May 2017].
- [27] Angular, "Angular Style Guide," [Online]. Available: <https://angular.io/styleguide>. [Accessed 21 May 2017].
- [28] Planespotters, "Planespotters.net," 05 May 2017. [Online]. Available: <https://www.planespotters.net/airframe/McDonnell-Douglas/DC-9/47303/N952VV-Valujet-Airlines>.
- [29] estimote, "estimote," 05 May 2017. [Online]. Available: <https://estimote.com/>.
- [30] NativeScript, "Use Native Libraries," [Online]. Available: <https://docs.nativescript.org/angular/runtimes/ios/how-to/Use-Native-Libraries.html>. [Accessed 14 March 2017].
- [31] I. Buhov, "GitHub," [Online]. Available: <https://github.com/NativeScript/NativeScript/issues/1192>. [Accessed 13 March 2017].
- [32] Y. Wang, X. Yang, Y. Zhao, Y. Liu and L. Cuthbert, "Bluetooth positioning using RSSI and triangulation methods," in *Bluetooth positioning using RSSI and triangulation methods*, Las Vegas, 2013.
- [33] Y. Wang, X. Yang, Y. Zhao, L. Cuthbert and Y. Liu, "Bluetooth Positioning using RSSI and Triangulation," in *2013 IEEE 10th Consumer Communications and Networking Conference*, Las Vegas, 2013.
- [34] O. Wenxing, W. Lei, J. Zhipeng and Y. Changhong, "Implementation of Smart Shopping System based on NFC Technology," in *2015 Seventh International Conference on Measuring Technology and Mechatronics Automation*, 2015.
- [35] K. Sletten, "UiO Research Archive," [Online]. Available:
-

- <https://www.duo.uio.no/handle/10852/51251?show=full>. [Accessed 20 May 2017].
- [36] V. K. Varma, "Wireless Fidelity—WiFi," in *Originally published on the IEEE Emerging Technology portal, 2006 - 2012..*
- [37] A. Mahadik, Y. Katta, R. Naik, N. Naikwade and N. F. Shaikh, "A review of Augmented Reality and its application in context aware library system," in *ICT in Business Industry & Government (ICTBIG), International Conference on*, Indore, India, India, 18-19 Nov. 2016.
- [38] o. dictionaries, "Definition Augmented Reality".
- [39] Estimote, "Estimote.com," [Online]. Available: <http://estimote.com/#get-beacons>. [Accessed 07 April 2017].
- [40] Kontakt, "Kontakt.io," [Online]. Available: <https://store.kontakt.io/next-generation/32-beacon-pro.html>. [Accessed 07 April 2017].
- [41] Progress, "GitHub," [Online]. Available: <https://github.com/progress/estimote-nativeScript-plugin>. [Accessed 11 April 2017].
- [42] T. Z. Warfel, Prototyping, Rosenfeld Media, 2009, pp. 16-17.
- [43] "Creating App Icons and Launch Screens for iOS," 20 April 2017. [Online]. Available: <http://docs.nativescript.org/angular/publishing/creating-launch-screens-ios.html>.
- [44] "About App Distribution Workflow," [Online]. Available: <https://developer.apple.com/library/content/documentation/IDEs/Conceptual/AppDistributionGuide/Introduction/Introduction.html>. [Accessed 20 April 2017].
- [45] "Configuring Your Xcode Project for Distribution," [Online]. Available: <https://developer.apple.com/library/content/documentation/IDEs/Conceptual/AppDistributionGuide/ConfiguringYourApp/ConfiguringYourApp.html>. [Accessed 20 April 2017].
- [46] "Uploading Your App to iTunes Connect," [Online]. Available: <https://developer.apple.com/library/content/documentation/IDEs/Conceptual/AppDistributionGuide/UploadingYourAppToiTunesConnect/UploadingYourAppToiTunesConnect.html>. [Accessed 20 April 2017].

-
- [47] [Online]. Available: https://developer.apple.com/library/content/documentation/LanguagesUtilities/Conceptual/iTunesConnect_Guide/Chapters/SubmittingTheApp.html. [Accessed 20 April 2017].
- [48] "developer.android.com," [Online]. Available: <https://developer.android.com/design/index.html>. [Accessed 20 April 2017].
- [49] [Online]. Available: <http://docs.nativescript.org/publishing/publishing-android-apps>. [Accessed 20 April 2017].
- [50] [Online]. Available: <http://www.telerik.com/blogs/hybrid-or-native-mobile-app-use-the-right-tool-for-the-job>. [Accessed 20 April 2017].
- [51] "Creating App Icons and Launch Screens for iOS," [Online]. Available: <http://docs.nativescript.org/publishing/creating-launch-screens-ios>. [Accessed 20 April 2017].
- [52] K. Saranya, R. Reminaa and S. Subhitha, "Modern applications of QR-Code for security," 2016.
- [53] R. Want, "An introduction to RFID technology," *IEEE Pervasive Computing*, vol. 5, 13 February 2006.
- [54] Technovelgy LLC, [Online]. Available: <http://www.technovelgy.com/ct/technology-article.asp>. [Accessed 20 april 2016].
- [55] F. Stroud, "What Is a Beacon? Webopedia Definition," [Online]. Available: <http://www.webopedia.com/TERM/B/beacon.html>. [Accessed 07 April 2017].
- [56] Telerik, "Execution Flow," [Online]. Available: <https://docs.nativescript.org/runtimes/android/advanced-topics/execution-flow>. [Accessed 07 April 2017].
- [57] K. Townsend, C. Cufi, Akiba and R. Davidson, *Getting Started with Bluetooth Low Energy*, O'Reilly Media, Inc., 2014.
- [58] Telerik, "Creating launch screens android," [Online]. Available: <https://docs.nativescript.org/publishing/creating-launch-screens-android>. [Accessed
-

20 April 2017].

- [59] Apple, "iOS Human Interface Guidelines," [Online]. Available: <https://developer.apple.com/ios/human-interface-guidelines/overview/design-principles/>. [Accessed 20 April 2017].
- [60] "Technical Q&A QA1686 App Icons on iPhone, iPad and Apple Watch," Apple, 20 Desember 2016. [Online]. Available: https://developer.apple.com/library/content/qa/qa1686/_index.html. [Accessed 20 April 2017].
- [61] "Units," 2016. [Online]. Available: <http://softwaretestingfundamentals.com/unit-testing/>. [Accessed 09 May 2017].
- [62] M. Rouse, "unit testing," 2006-2007. [Online]. Available: <http://searchsoftwarequality.techtarget.com/definition/unit-testing>. [Accessed 09 May 2017].
- [63] Atlassian, "Viewing the Burndown Chart," [Online]. Available: <https://confluence.atlassian.com/agile/jira-agile-user-s-guide/using-a-board/using-reports/viewing-the-burndown-chart>. [Accessed 09 May 2017].
- [64] P. Maltez. [Online]. Available: <https://github.com/pedromaltez/typescript-logo-recreation>. [Accessed 18 May 2017].
- [65] indigitall, [Online]. Available: <https://indigitall.net/inicio/beacons/>. [Accessed 18 May 2017].
- [66] "Eight to Late," [Online]. Available: <https://eight2late.wordpress.com/2009/07/01/cox%E2%80%99s-risk-matrix-theorem-and-its-implications-for-project-risk-management/>. [Accessed 18 May 2017].
- [67] Bluetooth, 05 05 2017. [Online]. Available: <https://www.bluetooth.com/>.
- [68] P. Krill, "JavaScript goes native for iOS, Android, and Windows Phone apps," InfoWorld, 10 March 2015. [Online]. Available: <http://www.infoworld.com/article/2893706/mobile-development/javascript-native-ios-android-and-windows-phone-apps.html>. [Accessed 07 April 2017].

-
- [69] Wikipedia, "Bluetooth Low Energy," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Bluetooth_low_energy. [Accessed 20 April 2017].
- [70] O. Dictionary, "Oxford Dictionary," 2017. [Online]. Available: <https://en.oxforddictionaries.com/definition/beacon>.
- [71] R. Costa, "Linkedin," 02 10 2016. [Online]. Available: <https://www.linkedin.com/pulse/who-where-what-when-why-how-use-scrum-simple-guide-rui-costa>. [Accessed 05 May 2017].
- [72] Estimote, "Location Beacon," [Online]. Available: <http://estimote.com/assets/gfx/press/product/Location-Beacon-Pink-Background.dd5138d6.png>. [Accessed 07 April 2017].
- [73] D. Ramel, "Telerik's NativeScript Aims To Centralize Cross-Platform Mobile Development," 03 September 2015. [Online]. Available: <https://visualstudiomagazine.com/articles/2015/03/09/telerik-nativescript-beta-mobile.aspx>. [Accessed 07 April 2017].
- [74] T. Skjetne, "UiO Research Archive," [Online]. Available: <https://www.duo.uio.no/bitstream/handle/10852/51097/1/Torkjel-Skjetne-Masteroppgave-mai-2016.pdf>. [Accessed 20 May 2017].

APPENDIX A – USER MANUAL

INSTALLING THE APPLICATION

If you want to install the application yourself on your mobile phone or tablet to test it out. We have written a short guide on how to do this.

IOS

Requirements:

- iPhone 4s+
- iOS 7+
- iTunes

IOS INSTALLATION INSTRUCTION

Step -01

- Open iTunes and click on View => Media Kind => Apps.

Step-02

- Drag the file provided with the project “beacon.ipa”, into the Apps section in iTunes.

Step-03

- Connect the iPhone to the computer.
- Click on the iPhone icon in the navigation bar in iTunes
- Click on App and click install beacon then click on apply

This should install the application to your iPhone or iPad. When you open the application for the first time, it will ask you for permission to use location services/Bluetooth. For the beacons part of the application to work you need to allow this.

ANDROID

Requirements:

- Android phone
- Android 4+
- Google account

ANDROID INSTALLATION INSTRUCTION

Step-01

- Go into setting => Security and allow installation from unknown sources.

Step-02

- Transfer the apk file provided as an attachment to the phone

Step-03

- Click on the beacon.apk file and wait for it to download and click install when prompted if you want to install the app.

USING THE APPLICATION

This is a manual describing how to use the beacon application. When starting up the application you will get a view of the loaded publication.

TABLE OF CONTENTS VIEW

The middle of the screen shows the Toc view. Here the table of contents of the data set is displayed and you can click on the different sections of the document to navigate. When you reach a document, it will open the document.

On the top right corner is the button “Beacon” to access the beacons. When you click this button, you will be redirected to a list of beacons. When the button is red it means that beacons have been detected. If the button is blue there is no available beacons nearby.

If you want to go back you can click the arrow button in the top left corner. This is a backspace and redirect you to the last page you were at.

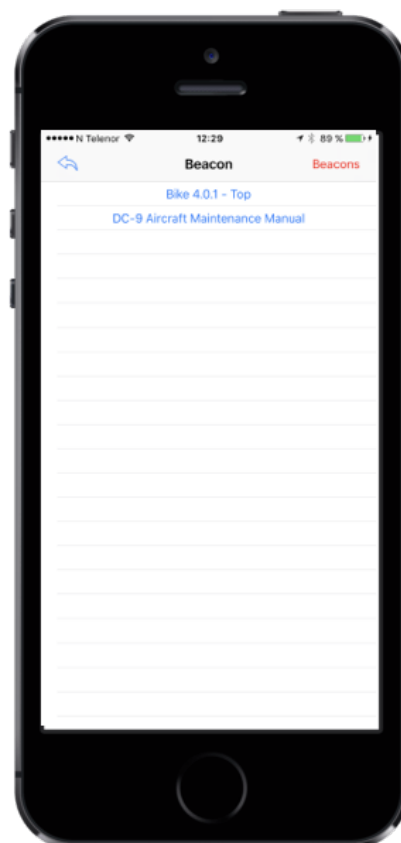


Figure 38 - Toc view of bike data set

BEACON VIEW

In the beacon view you can choose which beacon applicability you want. You select the beacon by swiping your finger up or down. Make sure the Selected applicability is the right one. When you have selected the right beacon, you can click “Apply” and the application will redirect you to where you were before.

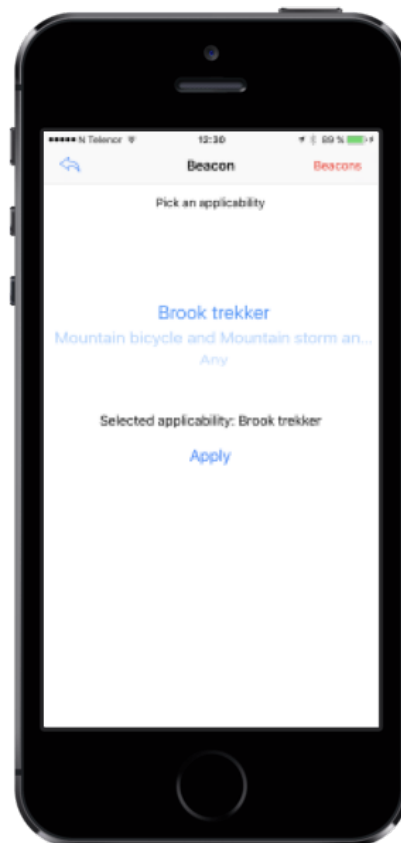


Figure 39 - Beacon view

DOCUMENT VIEW

This is the document view and will display the selected document. We have selected the Mountain Bicycle Mountain storm MK1 beacon here and the document has filtered out the none applicable Mountain Bicycle Brook trekker Mk9. To change the applicability, you can click on the “beacons” button. In this view, you can swipe through documents. To do this you swipe your finger to the left or right.

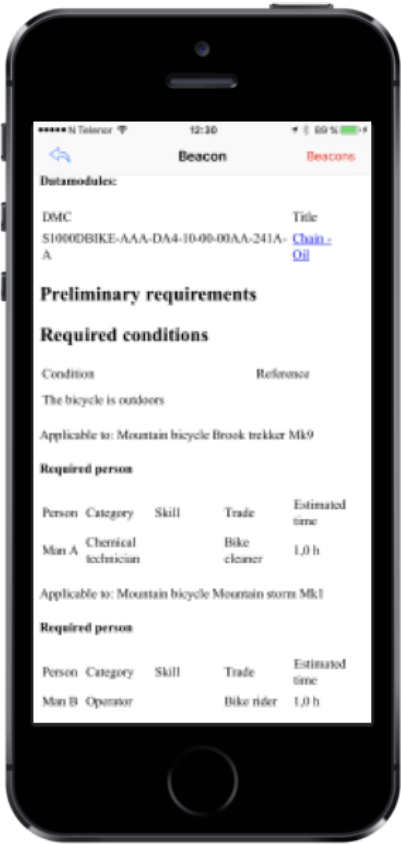


Figure 40 - Document view



Figure 41 - Document view (Filter Applied)

APPENDIX B – DECISIONS

This is a document where we discuss the major decisions we have taken during the project.

NATIVESCRIPT DECISION

Flatirons Solutions recommended that we use NativeScript to develop the prototype since this was very similar to what they are using. After some research, we found that this would be good to use since it required minimal work to quickly get an application up and running.

BEACON DECISION

When looking for possible beacons we narrowed it down to either Kontakt or Estimote. This is due to them having an SDK for their beacons, and the support offered after purchasing the beacons. This SDK gives us interfaces and extra features when developing using beacons. We have collected information regarding Estimote and Kontakt beacons. We have put the information inside a table you can see underneath.

| | Estimote Location Beacon [38] | Kontakt Beacon Pro [39] |
|----------------------------|------------------------------------------------------------|------------------------------------------------------------------------------|
| Range | 200M | 80M |
| Battery Life | 5 Years | 5 Years |
| Packets | iBeacon, Eddystone, Telemetry, User defined | iBeacon and Eddystone |
| Built-in sensors | Motion, Temperature, ambient light, Magnetometer, Pressure | Accelerometer, Ambient light, |
| Additional features | Mesh networking, GPIO, RTC, LED, 1Mb EEPROM | Real time clock, LED indicator, Optional USB communication, NFC, Water Proof |
| Price | 99\$ for 3 | 87\$ for 3 |

Using the table above we can quickly see that Estimote and Kontakt both have much in common. They both meet the basic requirement of being able to transmit iBeacon packages. What made us choose Estimote instead of Kontakt or any other beacon was the possibility to use indoor location. This is something we initially wanted to try out, but we found this was not possible through their API with only 4 beacons. In the future work section in chapter 9, we have a more detailed description on how we believe indoor location could work with the S1000D specification. This combined with the fact that Estimote has a heavy focus on making it easy for developers to get started.



Figure 42 - Estimote Beacons

ESTIMOTE PLUGIN DECISION

Before we started developing we had considered solutions to integrate beacons into NativeScript. During our research, we found someone that developed a plugin that implements the EstimoteSDK [40]. The problem was that this was a plugin that has been abandoned. It has an old version of the EstimoteSDK and it is not updated for the newer NativeScript versions. We discussed if we would take the time to update the plugin or if we should just start from scratch and base our plugin on this one. Ultimately, we fell on making our own. The arguments for making our own was:

- Typescript support
- Nearly no documentation of how the plugin worked
- The learning outcome will be handy to develop new features for beacons if we have time at the end of the project
- Newest version of EstimoteSDK.

The main argument against making our own plugin was that it would take extra time. We estimated that the extra time was worth it. The experience we would get will be important and we will have a much better understanding of how EstimoteSDK and NativeScript plugins work.

APPENDIX C – SPRINT

SPRINT 1

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (- → 2) |
|-------------|--------------------------------|------------|-----------|----------|----------------------|
| BEACON-1 | Project plan | Epic | ↑ High | RESOLVED | - |
| BEACON-7 | Define Roles | Story | ↑ Medium | CLOSED | - |
| BEACON-8 | Understand the project | Task | ↑ Highest | CLOSED | - |
| BEACON-12 * | Define Project Scope | Story | ↑ Medium | CLOSED | - |
| BEACON-13 * | Research about beacon | Story | ↑ Medium | CLOSED | - → 2 |
| BEACON-14 * | Group Image | Story | ↑ Medium | CLOSED | - |
| BEACON-23 * | Converting Wiki pages to latex | Story | ↑ Medium | CLOSED | - |

Figure 43 - Sprint 1 issues

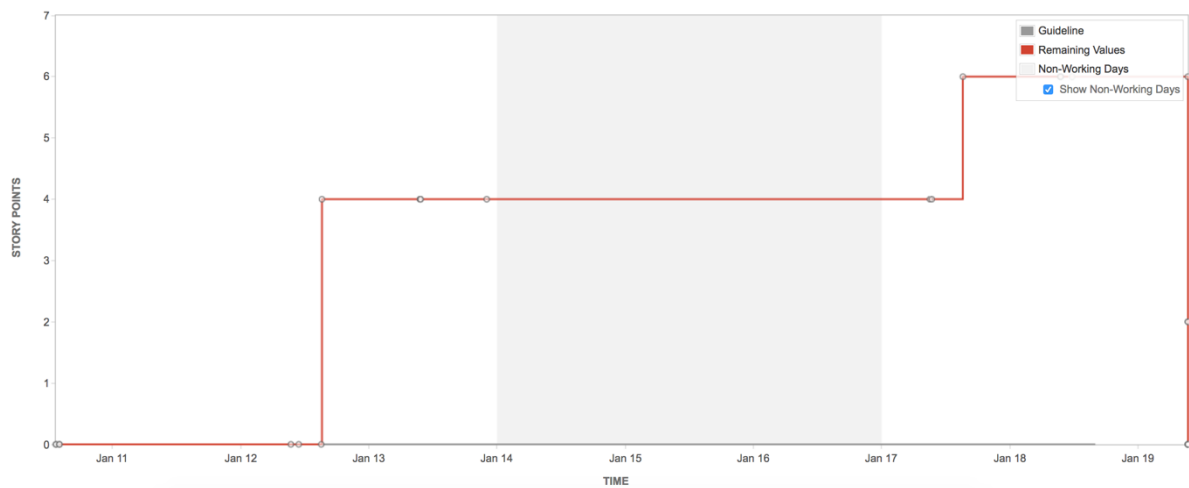


Figure 44 - Sprint 1 burndown

2017.01.19 Retrospective Sprint 1

Date: 19.01.2017

Participants: @Daniel, Tor Egil Øverberg, Ole Christian Moholth, @Afshin

What did we do well?

- Managed to get started with JIRA quickly
- Started before we had our first meeting with our advisor
- We've been active planning

What should we have done better?

- Set a sprint goal
- Value estimate backlog items
- Prepare backlog before sprint
- Log work on stories.
- Make sure there is enough work in this active sprint
- Plan sprint

Impediments

- Everyone lacked experience. Didn't really know what to do.
- We lacked information about our supervisor
- Hard to get in contact with Olaf

Actions

- ✓ None at the moment

Figure 45 - Sprint 1 retrospective

SPRINT 2

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (49) |
|-----------------------------|---------------------------------|------------|----------|--------|-------------------|
| BEACON-6 | Presentation | Story | Medium | CLOSED | 13 |
| BEACON-24 | Create invitation and send out. | Story | Medium | CLOSED | 2 |
| BEACON-25 | Project Model | Story | Medium | CLOSED | 5 |
| BEACON-39 | LaTeX Document | Story | Medium | CLOSED | 8 |
| BEACON-40 | Rewrite project description | Story | Medium | CLOSED | 8 |
| BEACON-43 | Risk Analysis and Management | Story | Medium | CLOSED | 13 |
| BEACON-44 * | Select type of beacon | Story | Medium | CLOSED | - |
| BEACON-46 * | Test analyse etc | Story | Medium | CLOSED | - |
| BEACON-48 * | Write user stories | Story | Medium | CLOSED | - |
| BEACON-49 * | Technical requirements | Story | Medium | CLOSED | - |
| BEACON-50 * | UML use case | Story | Medium | CLOSED | - |

Figure 46 - Sprint 2 issues

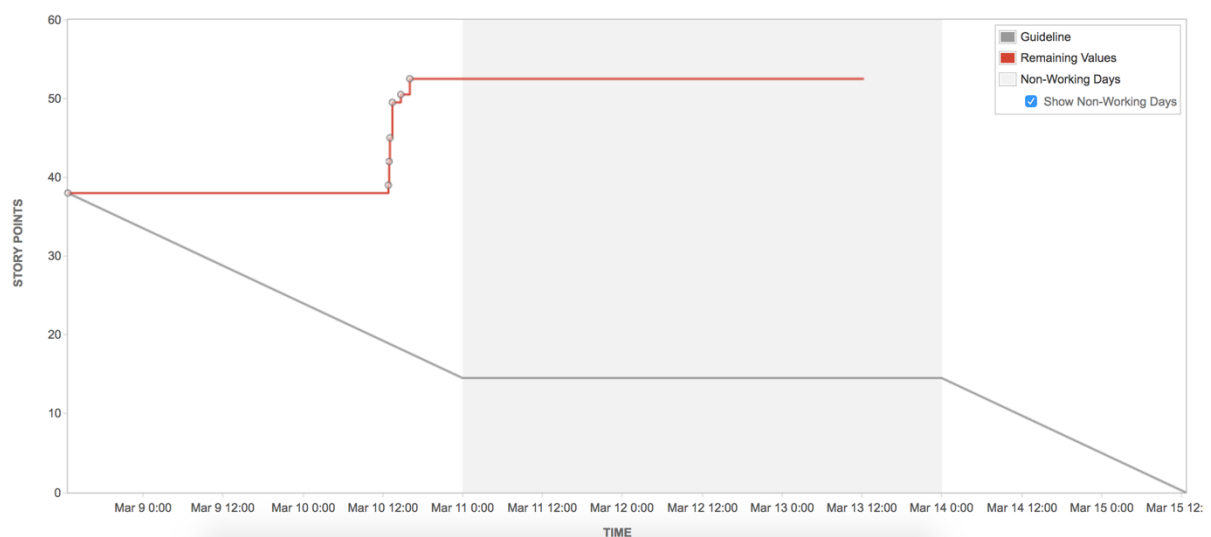


Figure 47 - Sprint 2 burndown

2017.02.01 Retrospective Sprint 2

Date: 01.02.2017

Participants: @Daniel, Tor Egil Øverberg, Ole Christian Moholth, @Afshin

What did we do well?

- Presentation went well
- We managed to do a lot

What should we have done better?

- Get better at making user stories
- Generally need better documents
- Test and test document
- Include References
- get better at logging hours to user stories
- Get better at using Jira

Impediments

- A little harder to get in contact with Radmila.

Actions

- ☐ look at user stories and create test.
- ☐

Figure 48 - Sprint 2 Retrospective

SPRINT 3

Completed Issues

View in Issue Navigator

| Key | Summary | Issue Type | Priority | Status | Story Points (61) |
|---------------------------|--------------------------------------------------|------------|----------|--------|-------------------|
| BEACON-58 | Understand pinpoint neutral package format | Story | Medium | CLOSED | 40 |
| BEACON-61 | get started with nativescript | Story | Highest | CLOSED | 20 |
| BEACON-65 | Fix repository / make root project / name beacon | Story | Medium | CLOSED | 1 |

Figure 49 - Sprint 3 Issues

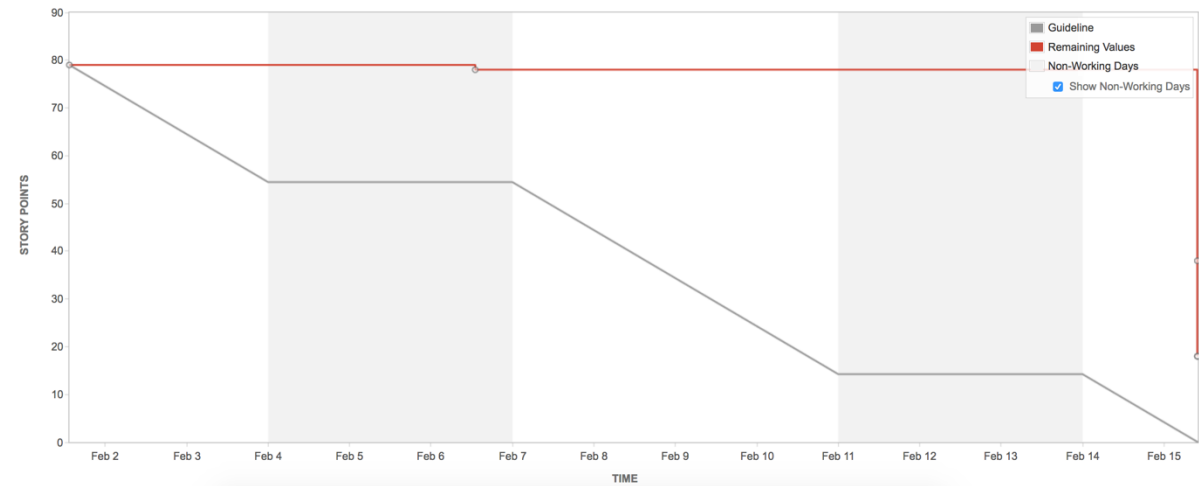


Figure 50 - Sprint 3 Burndown

2017.02.15 Retrospective Sprint 3

Date: 15.02.2017

Participants: @Daniel, Tor Egil Øverberg, Ole Christian Moholth, @Afshin

What did we do well?

- Quickly got a working document viewer.

What should we have done better?

- Document more of everything we do
- Writing tests
- Log work in JIRA

Impediments

- Lachlan week. Lost one week work
- Lack of experience

Actions

- ☐ Ole, Tor and Daniel need to meet on time.

Figure 51 - Sprint 3 Retrospective

SPRINT 4

Completed Issues

View in Issue Navigator

| Key | Summary | Issue Type | Priority | Status | Story Points (18.5) |
|-----------|------------------------|------------|----------|--------|---------------------|
| BEACON-60 | Show table of contents | Story | Highest | CLOSED | 13 |
| BEACON-62 | Make website | Story | Medium | CLOSED | 5 |
| BEACON-76 | Purchase beacons | Story | Highest | CLOSED | 0.5 |

Figure 52 - Sprint 4 Issues

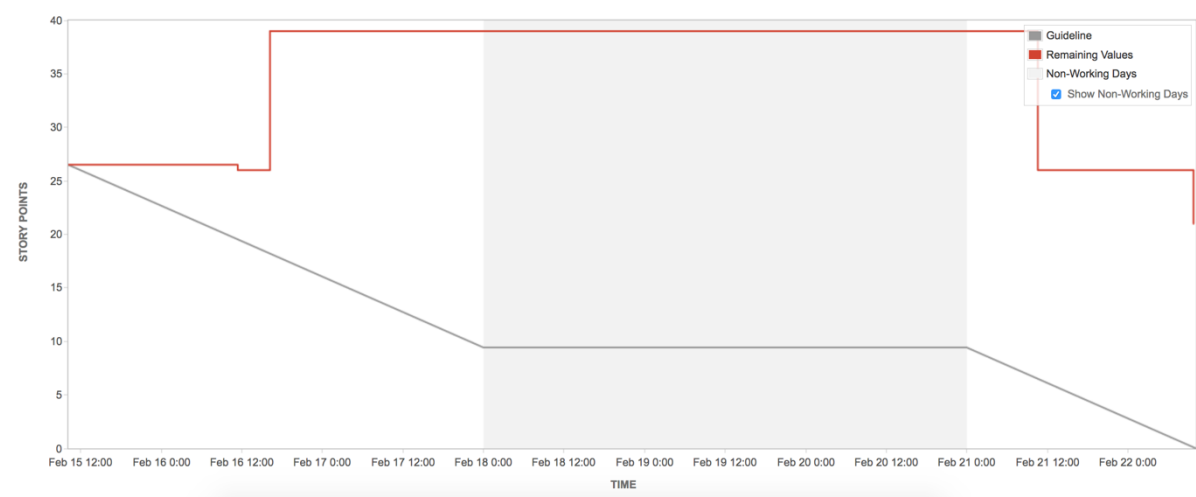


Figure 53 - Sprint 4 Burndown

2017.02.22 Retrospective Sprint 4

Date: 15.02.2017

Participants: @Daniel, Tor Egil Øverberg, Ole Christian Moholth, @Afshin

What did we do well?

- Good website
- Table of Contents
- Better at logging hours in JIRA

What should we have done better?

- Better at documentation
- Break down stories better. Make sure we have a story/task for everything.
- Log all work onto a JIRA subtask
- Unit testing. (Test report)

Impediments

Actions

Figure 54 - Sprint 4 Retrospective

SPRINT 5

Issues Not Completed

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (28) |
|-----------|-------------------------|------------|----------|-------------|-------------------|
| BEACON-74 | Filter by applicability | Epic | Highest | IN PROGRESS | 20 |
| BEACON-79 | Beacons communication | Story | High | UNASSIGNED | 8 |

Figure 55 - Sprint 5 Issues

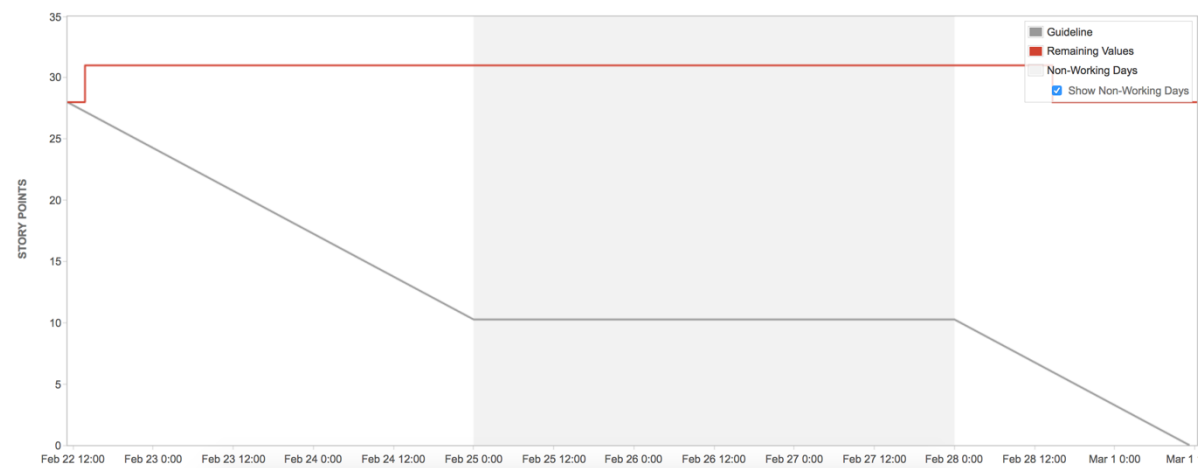


Figure 56 - Sprint 5 Burndown

2017.03.01 Retrospective Sprint 5

Participants: @Daniel, @Tor Egil, @Ole Christian, @Afshin

What did we do well?

- We're better at logging hours
- Better at decomposing stories
- We're showing up when we're supposed to

What should we have done better?

- Subtasks should have been stories. Burndown is at a standstill due to not completed storeis
- Include the entire team
- We need better descriptions of subtasks
- We need to get better at documentation
- We need to spend more time documenting work

Impediments

Actions

- Put more stories in the worklog. We need to have more stories than just the next sprint

Figure 57 - Sprint 5 Retrospective

SPRINT 6

Completed Issues

View in Issue Navigator

| Key | Summary | Issue Type | Priority | Status | Story Points (42 → 50.5) |
|--------------|---------------------------------|------------|----------|--------|--------------------------|
| BEACON-88 * | Filter html components | Story | Medium | CLOSED | 13 → 8 |
| BEACON-92 * | Uniforms | Story | High | CLOSED | - → 0.5 |
| BEACON-94 * | Plan Application view structure | Story | Medium | CLOSED | 3 |
| BEACON-112 * | Integrate ios EstimateSDK | Story | Medium | CLOSED | 13 |
| BEACON-114 * | Detect beacons | Story | Medium | CLOSED | - → 13 |
| BEACON-115 * | Refine the product backlog | Story | Medium | CLOSED | 13 |

Figure 58 - Sprint 6 Issues

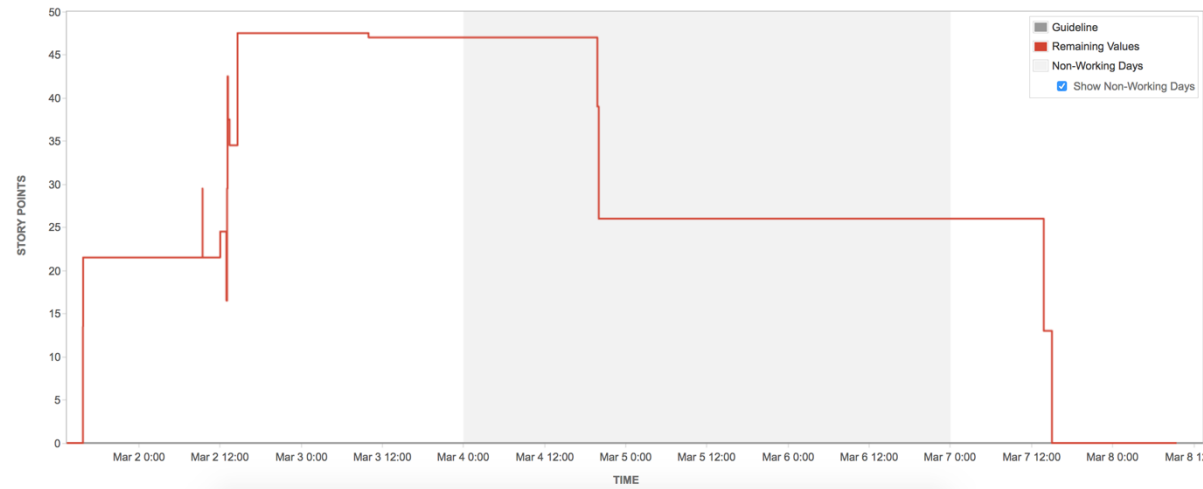


Figure 59 - Sprint 6 Burndown

2017.03.08 Retrospective Sprint 6

What did we do well?

- Finished all stories.
- Break down user stories to more manageable tasks
- Good at logging hours on JIRA issues/epics
- Did backlog refinement

What should we have done better?

- Documentation

Impediments

Actions

Figure 60 - Sprint 6 Retrospective

SPRINT 7

Completed Issues

View in Issue Navigator

| Key | Summary | Issue Type | Priority | Status | Story Points (41.5) |
|--------------|----------------------------------------------------------|------------|----------|--------|---------------------|
| BEACON-93 | Filter UI | Story | Medium | CLOSED | 8 |
| BEACON-122 | Plugin Documentation | Story | Medium | CLOSED | 5 |
| BEACON-123 | User stories latex | Story | Medium | CLOSED | 5 |
| BEACON-124 | Applicability Documentation | Story | Medium | CLOSED | 3 |
| BEACON-125 | Toc Documentation | Story | Medium | CLOSED | 2 |
| BEACON-126 | Document view (webview) | Story | Medium | CLOSED | 3 |
| BEACON-128 | Document preparation (HTML builder) | Story | Medium | CLOSED | 1 |
| BEACON-131 | Update budget | Story | Medium | CLOSED | 0.5 |
| BEACON-132 | Improve risk document | Story | Medium | CLOSED | 2 |
| BEACON-133 | Reorganize LaTeX documents. | Story | Medium | CLOSED | 0.5 |
| BEACON-138 * | Routing Documentation | Story | Medium | CLOSED | 1 |
| BEACON-139 * | Update application view structure with Radmila feedback. | Story | Medium | CLOSED | 3 |
| BEACON-140 * | Get Epics into latex documentation | Story | Medium | CLOSED | 3 |
| BEACON-141 * | Document how we communicate | Story | Medium | CLOSED | 0.5 |
| BEACON-143 * | Document previous sprints | Story | Medium | CLOSED | 3 |
| BEACON-144 * | Document how we detect beacons(OS part for now) | Story | Medium | CLOSED | 1 |

Figure 61 - Sprint 7 Issues

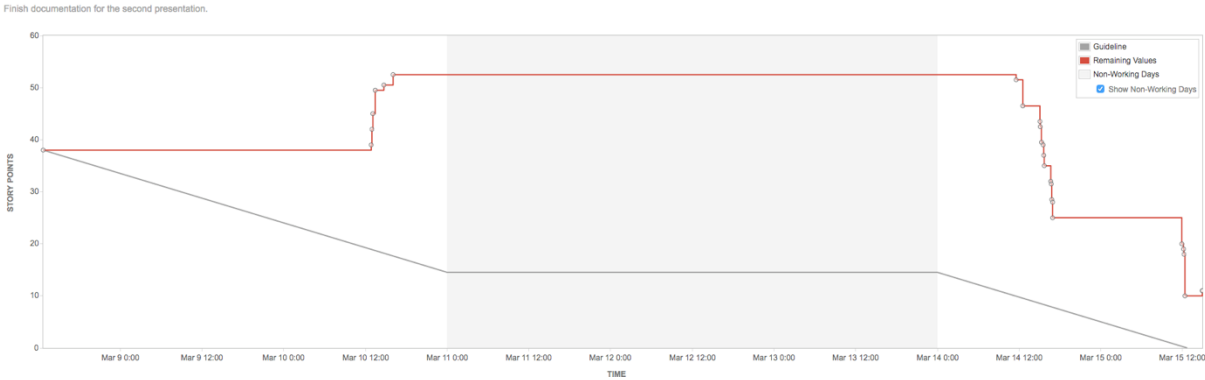


Figure 62 - Sprint 7 Burndown

2017.03.15 Retrospective Sprint 7

What did we do well?

- Smaller stories.
- We refined backlog .
- Overall a good sprint.

What should we have done better?

- Logging hours went south this week.
- Keep meetings shorter.
- Try not to stray off topic.

Impediments

Actions

Figure 63 - Sprint 7 Retrospective

SPRINT 8

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (51) |
|----------------------------|------------------------------------------------------------------------|------------|----------|--------|-------------------|
| BEACON-103 | Beacon List UI | Story | High | CLOSED | 8 |
| BEACON-105 | Get applicability filter from group or beacon | Story | Medium | CLOSED | 5 |
| BEACON-127 | Document system (document management) | Story | Highest | CLOSED | 5 |
| BEACON-130 | Improve project flatirons and pinpoint description(Chapter 1) | Story | Highest | CLOSED | 3 |
| BEACON-142 | Document purpose of project | Story | Highest | CLOSED | 1 |
| BEACON-145 | Fill in and check description and acceptance criteria of user stories. | Story | Highest | CLOSED | 2 |
| BEACON-150 | Where are we now and where are we going | Story | Highest | CLOSED | 2 |
| BEACON-151 | Proof read | Story | High | CLOSED | 5 |
| BEACON-152 | Import all figures and make sure all figures are labeled correctly | Story | High | CLOSED | 1 |
| BEACON-155 | Create Power Point | Story | Highest | CLOSED | 2 |
| BEACON-156 | Plan the presentation (Storyline) | Story | Highest | CLOSED | 3 |
| BEACON-157 | Rehearsal | Story | High | CLOSED | 13 |
| BEACON-158 | Merge branches for demo | Story | Medium | CLOSED | 1 |

Figure 64 - Sprint 8 Issues

Second presentation

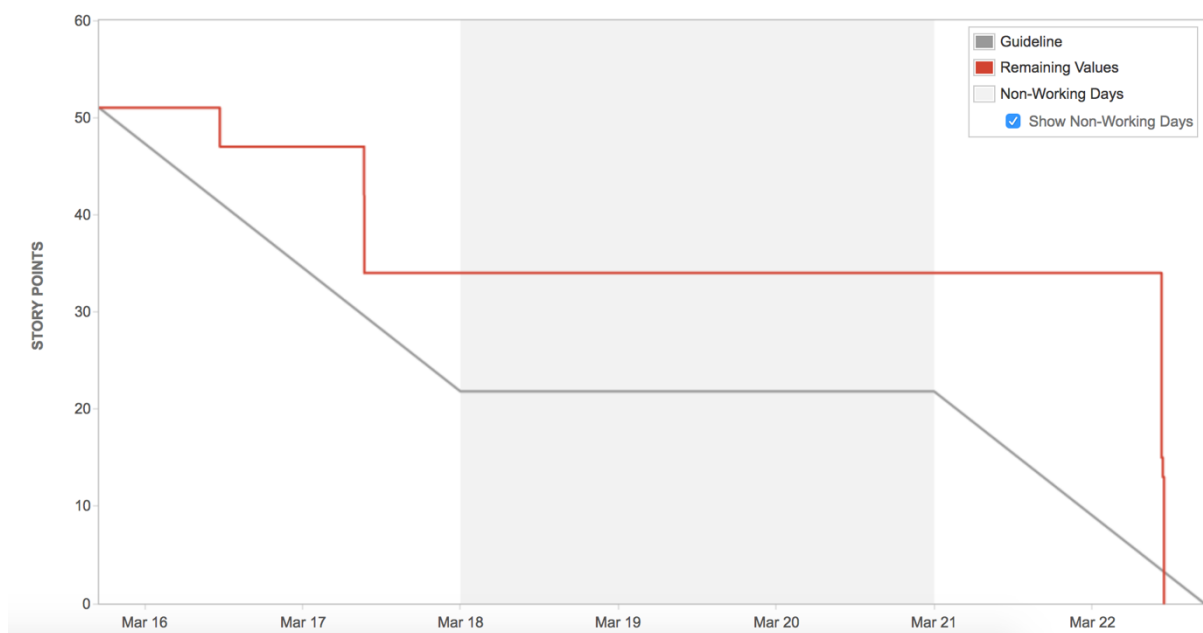


Figure 65 - Sprint 8 Burndown

2017.03.22 Retrospective Sprint 8

What did we do well?

- We had a good presentation
- Things went as expected

What should we have done better?

- More time with documentation
- Plan the documentation
- Have better understanding of where we are in the project
- Document any decisions we take

Impediments

Actions

- ☐ Document 2 hours each day.

Figure 66 - Sprint 8 Retrospective

SPRINT 9

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (5) |
|----------------------------|---------------------|-----------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------|---------------------|------------------|
| BEACON-159 | Plan future sprints |  Story |  Medium | CLOSED | 5 |

Figure 67 - Sprint 9 Issues

Plan the sprints Research the swipe functionality for Nativescript Detect beacons on Android

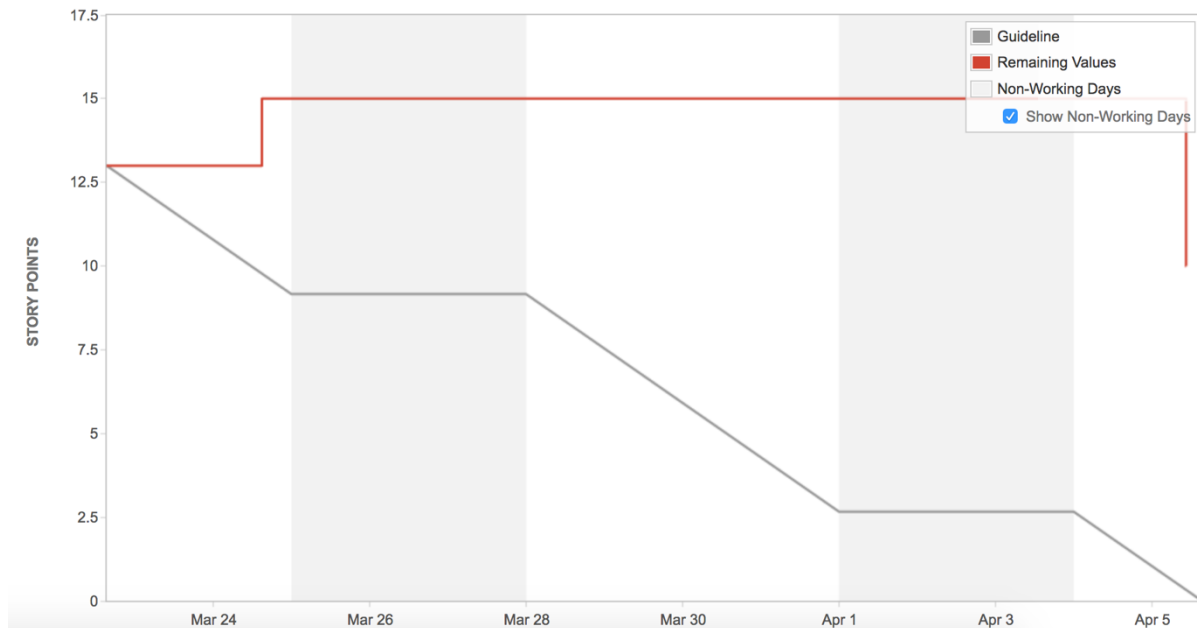


Figure 68 - Sprint 9 Burndown

2017.04.05 Retrospective Sprint 9

What did we do well?

What should we have done better?

- Not started this sprint

Impediments

- Web Science exam

Actions

- ☐ Document 2 hours each day.

Figure 69 - Sprint 9 Retrospective

SPRINT 10

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (32) |
|----------------------------|-----------------------------------------|------------|----------|--------|-------------------|
| BEACON-149 | DC9 data set | Story | High | CLOSED | 2 |
| BEACON-160 | Detect beacons on android | Story | Highest | CLOSED | 5 |
| BEACON-161 | Research swipe | Story | Medium | CLOSED | 3 |
| BEACON-164 | Restructure User stories section | Story | Highest | CLOSED | 8 |
| BEACON-169 | Write decision document Estimote Plugin | Story | Highest | CLOSED | 2 |
| BEACON-171 | Improve decision document for beacons | Story | High | CLOSED | 2 |
| BEACON-176 | Go to next document | Story | Highest | CLOSED | 8 |
| BEACON-177 | Code cleanup. | Story | Medium | CLOSED | 2 |

Figure 70 - Sprint 10 Retrospective

Implement beacon detection on Android. Start restructuring documents and implement swipe.

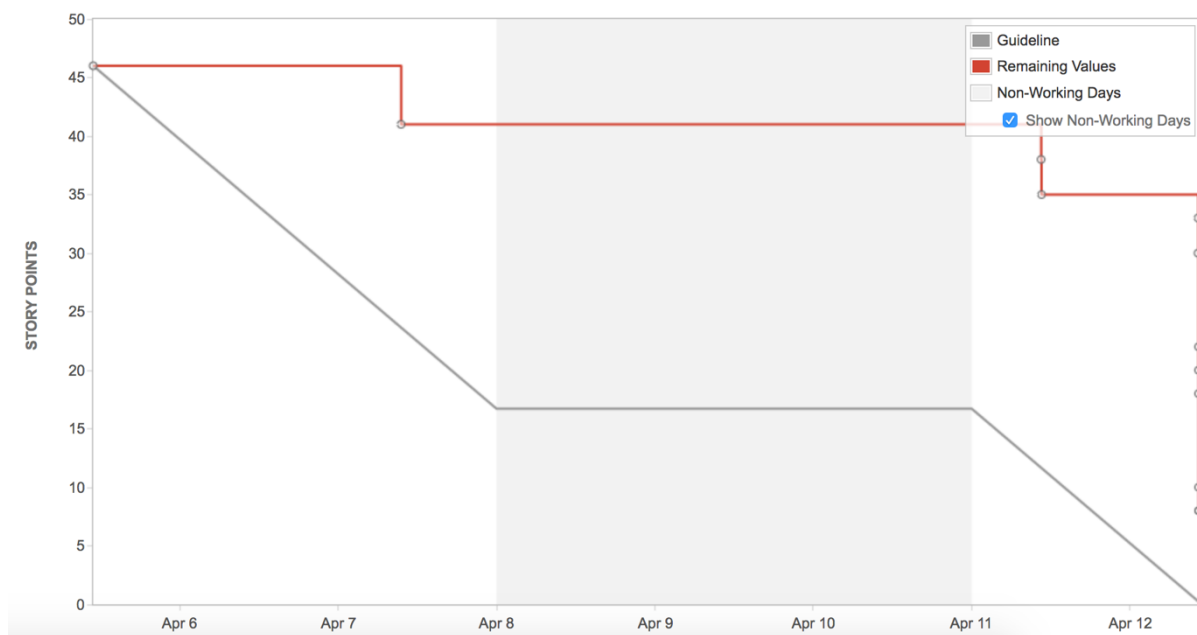


Figure 71 - Sprint 10 Burndown

2017.04.12 Retrospective Sprint 10

What did we do well?

- Better combination of documentation and coding

What should we have done better?

- Easter feeling. Low productivity some days.
- Started QA earlier in the week.

Impediments

- Easter feeling

Actions

Figure 72 - Sprint 10 Retrospective

SPRINT 11

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (20) |
|----------------------------|------------------------------------------------------------|------------|----------|--------|-------------------|
| BEACON-148 | Handle swipe gesture | Story | Highest | CLOSED | 5 |
| BEACON-163 | Plan initial documentation structure | Story | Highest | CLOSED | 3 |
| BEACON-170 | Write about beacon technology used with documentation | Story | High | CLOSED | 3 |
| BEACON-175 | Research generation of installation file from nativescript | Story | High | CLOSED | 1 |
| BEACON-181 | Notification | Story | Medium | CLOSED | 8 |

Figure 73 - Sprint 11 Issues

Easter sprint a little smaller than normal. Get swipe working on iOS, notification for beacon detection.

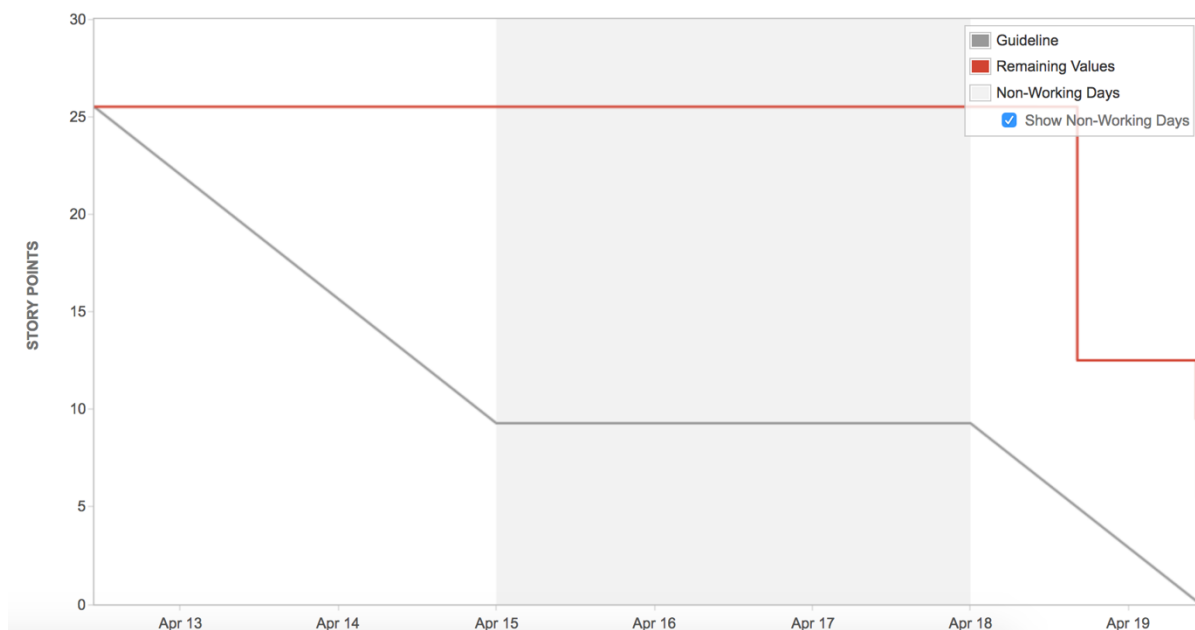


Figure 74 - Sprint 11 Burndown

2017.04.19 Retrospective Sprint 11

What did we do well?

- Finished all functionalities (Feature freeze)

What should we have done better?

- More documentation during easter
- More unit tests
- Get more feedback from Radmila

Impediments

- Easter

Actions

- ✓ Feature Freeze only bug fixes and documentation
- unit test everything

Figure 75 - Sprint 11 Retrospective

SPRINT 12

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (12.5 → 13.5) |
|------------------------------|------------------------------------------------------------------------------------|------------|----------|--------|----------------------------|
| BEACON-172 | Compare our solutions to other possible solutions | Story | Highest | CLOSED | 5 |
| BEACON-174 | Create User Manual for the application | Story | High | CLOSED | 3 |
| BEACON-184 | discuss location possibilities | Story | Medium | CLOSED | 1 |
| BEACON-185 | Future work draft | Story | Medium | CLOSED | 2 |
| BEACON-186 * | Import risk from latex to doc | Story | Medium | CLOSED | 0.5 |
| BEACON-187 * | Import project description from latex to doc | Story | Medium | CLOSED | 0.5 |
| BEACON-188 * | Import Process model from latex to doc | Story | Medium | CLOSED | 0.5 |
| BEACON-189 * | Fix comments from Radmila <small>Issue added to sprint after start time</small> | Story | Medium | CLOSED | - → 1 |

Figure 76 - Sprint 12 Issues

Document everything

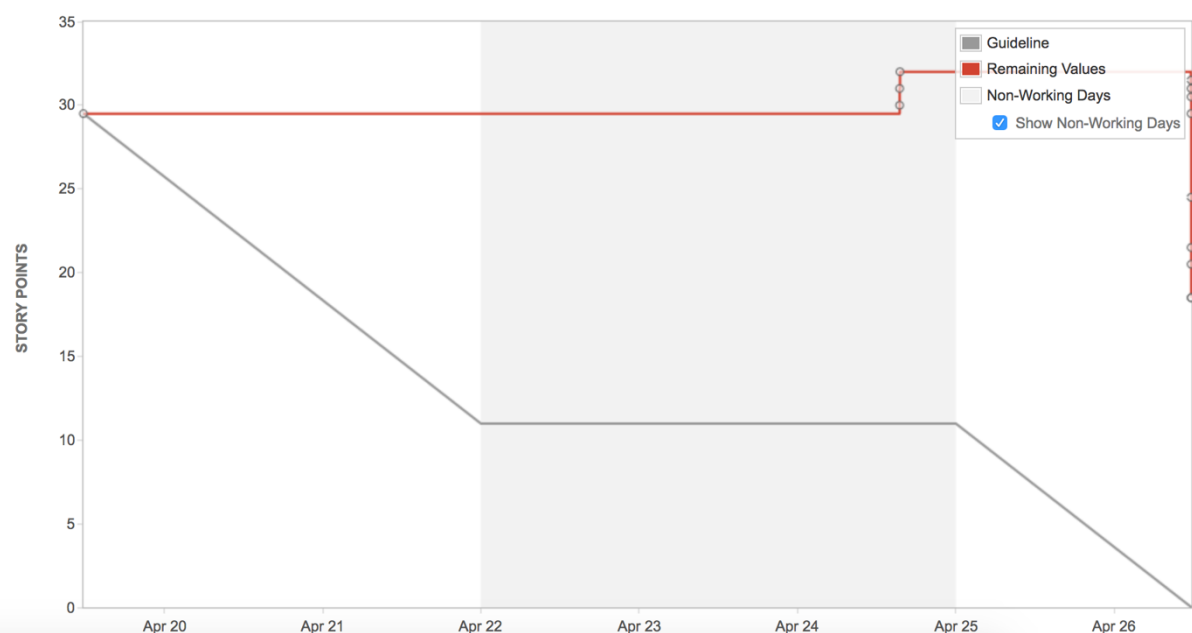


Figure 77 - Sprint 12 Burndown

2017.04.26 Retrospective Sprint 12

What did we do well?

- More contact with Radmila

What should we have done better?

- Bad planning.
- Estimated better.
- Work more efficiently.
- Remember description for issues.

Impediments

- Documentation has decelerated our velocity
- Daniel was sick on friday.

Actions

Figure 78 - Sprint 12 Retrospective

SPRINT 13

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (21 → 23) |
|------------------------------|-----------------------------------------------------------------------------------------------------------------|------------|----------|--------|------------------------|
| BEACON-182 | Research RSSI Filtering | Story | Medium | CLOSED | 5 |
| BEACON-190 | Discuss how we felt NativeScript has been. | Story | Medium | CLOSED | 2 |
| BEACON-192 | Improve manual for installing our application | Story | Medium | CLOSED | 1 |
| BEACON-193 | Try to arrange a interview with the person that have been a flight mechanic to see our application and feedback | Story | Medium | CLOSED | 3 |
| BEACON-195 | Estimate api usage | Story | Medium | CLOSED | 2 |
| BEACON-196 | Create publication manager unit test | Story | Medium | CLOSED | 2 |
| BEACON-197 | fix beacon-manager unit test | Story | Medium | CLOSED | 2 |
| BEACON-199 | Create publication model unit test | Story | Medium | CLOSED | 0.5 |
| BEACON-200 | Create document model unit test | Story | Medium | CLOSED | 0.5 |
| BEACON-201 * | Finish tables unit tests | Story | Highest | CLOSED | - → 2 |
| BEACON-202 * | Describe each unit test in detail | Story | Highest | CLOSED | 3 |

Figure 79 - Sprint 13 Issues

Documentation.

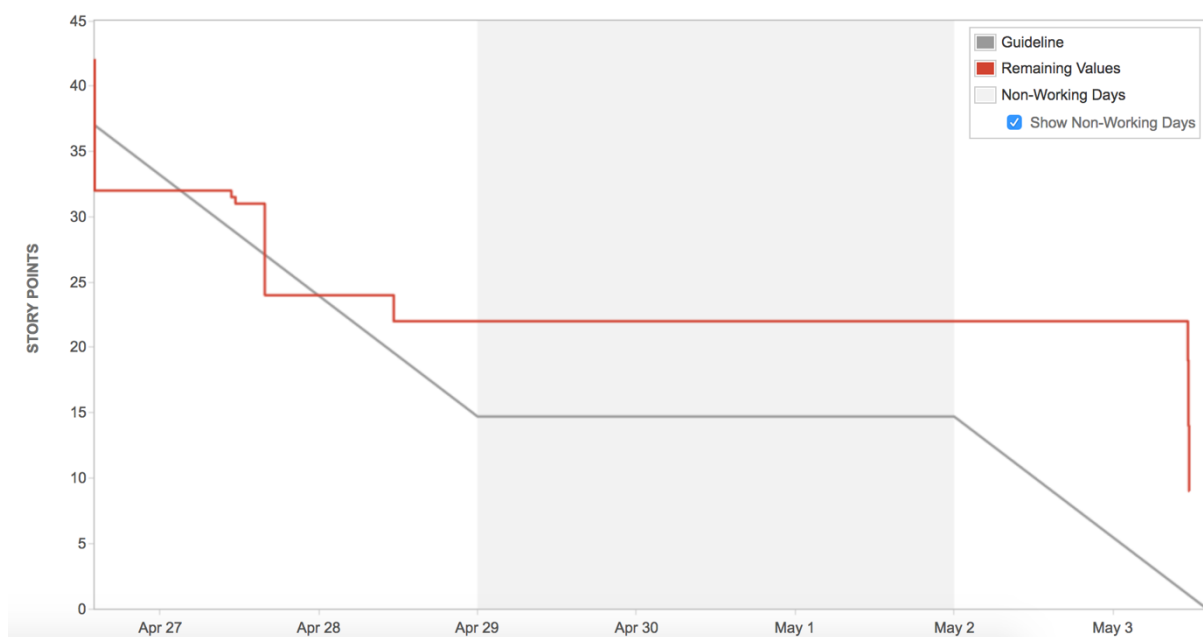


Figure 80 - Sprint 13 Burndown

2017.05.03 Retrospective Sprint 13

What did we do well?

- Sprint started well.
- On schedule

What should we have done better?

- More breaks.
- Work more efficiently.

Impediments

Actions





















- Define formal breaks and work hours.

Figure 81 - Sprint 13 Retrospective

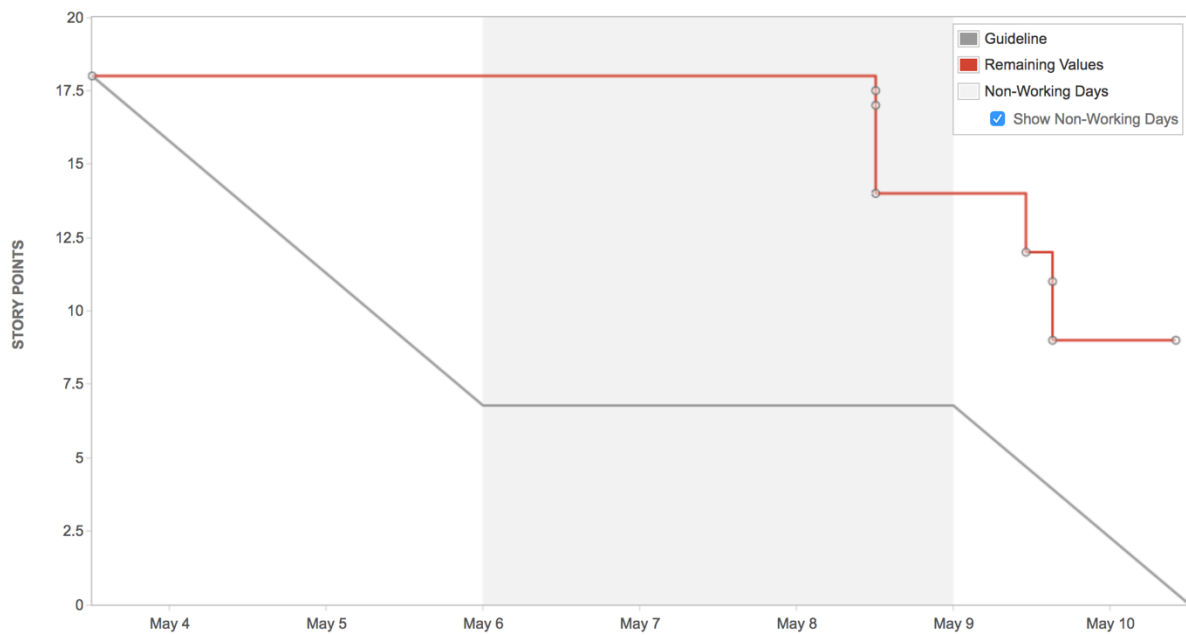
SPRINT 14

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (19) |
|----------------------------|-----------------------------------|-------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|---------------------|-------------------|
| BEACON-173 | Discuss how test has been used |  Story |  Highest | CLOSED | 1 |
| BEACON-194 | Publication manager documentation |  Story |  Medium | CLOSED | 3 |
| BEACON-203 | Describe our testing methods |  Story |  Highest | CLOSED | 2 |
| BEACON-206 | Future Work: Triangulation |  Story |  Medium | CLOSED | 1 |
| BEACON-207 | Find relevant pictures |  Story |  Medium | CLOSED | 2 |
| BEACON-212 | conclusion |  Story |  Medium | CLOSED | 3 |
| BEACON-213 | Diagrams for beacons |  Story |  Medium | CLOSED | 3 |
| BEACON-214 | ToC Component diagrams |  Story |  Medium | CLOSED | 2 |
| BEACON-215 | look into security chapter 9.3 |  Story |  Medium | CLOSED | 1 |
| BEACON-216 | Machine Learning |  Story |  Medium | CLOSED | 1 |

Documentation.



2017.05.10 Retrospective Sprint 14

What did we do well?

- On schedule
- Swipe fixed on android

What should we have done better?

- Estimation for documentation was wrong

Impediments

Actions

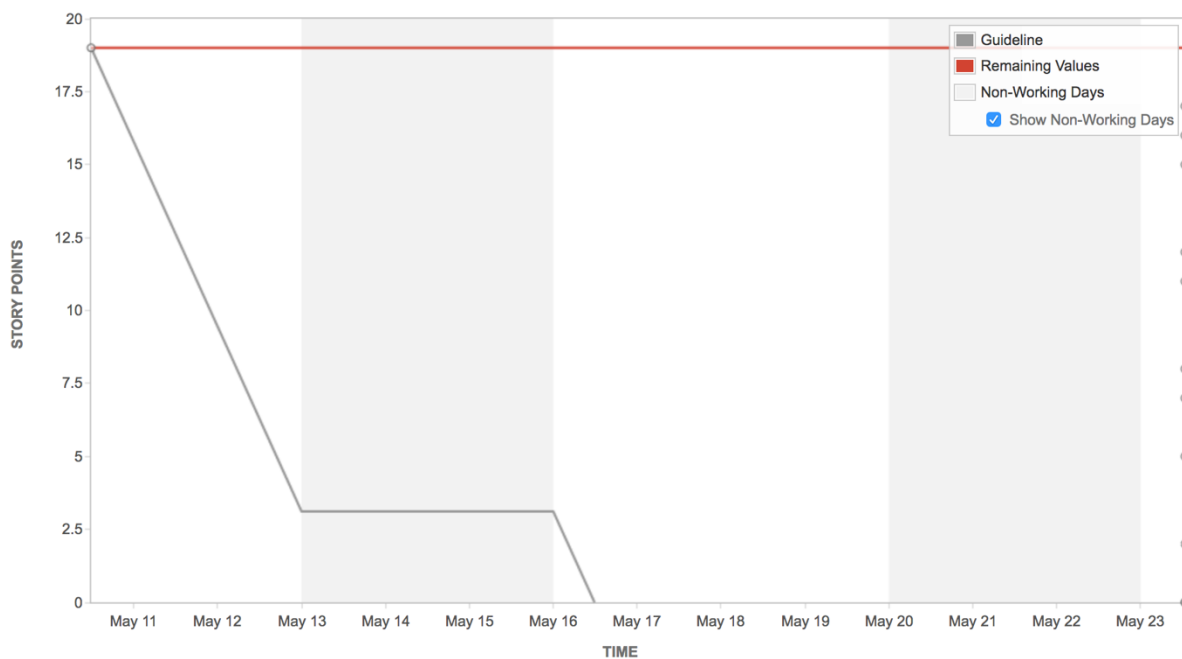
SPRINT 15

Completed Issues

[View in Issue Navigator](#)

| Key | Summary | Issue Type | Priority | Status | Story Points (19) |
|----------------------------|-----------------------------------|------------|----------|--------|-------------------|
| BEACON-173 | Discuss how test has been used | Story | Highest | CLOSED | 1 |
| BEACON-194 | Publication manager documentation | Story | Medium | CLOSED | 3 |
| BEACON-203 | Describe our testing methods | Story | Highest | CLOSED | 2 |
| BEACON-206 | Future Work: Triangulation | Story | Medium | CLOSED | 1 |
| BEACON-207 | Find relevant pictures | Story | Medium | CLOSED | 2 |
| BEACON-212 | conclusion | Story | Medium | CLOSED | 3 |
| BEACON-213 | Diagrams for beacons | Story | Medium | CLOSED | 3 |
| BEACON-214 | ToC Component diagrams | Story | Medium | CLOSED | 2 |
| BEACON-215 | look into security chapter 9.3 | Story | Medium | CLOSED | 1 |
| BEACON-216 | Machine Learning | Story | Medium | CLOSED | 1 |

Documentation



In sprint 15 we were only writing the documentation for our project. We felt that using stories to split the work load was unnecessary. This resulted in sprint 15 not being used properly.

APPENDIX D – TESTS

Most of the classes have unit tests to ensure correct functionality. In this section, all unit tests are described in the following fashion:

- A scenario describing preconditions, events and expected outcomes
- The section covered by the test, what functionalities are tested and the outcome of the test
- A table describing each unit test in detail. It covers what parameter is tested, the expected outcome and what assert is used to compare them.

PUBLICATION MANAGER

DOCUMENT FOLDER

SCENARIO

| | |
|--------------|-----------------------------------------------------------------------------|
| GIVEN | A folder with CORENA pinpoint documents exists |
| WHEN | A document folder is created |
| THEN | All CORENA pinpoint documents within the folder are accessible from id |
| AND | All CORENA pinpoint documents within the folder are accessible from pointer |

UNIT TESTS

| | | |
|-------------------------------|--------------------------------------------------------------------|--------|
| Create document folder | The list is created with the correct objects and evaluate function | Passes |
| Get by id | Documents can be retrieved from an id | Passes |
| Get by pointer | Documents can be retrieved from a pointer | Passes |

ASSERTS

CREATE DOCUMENT FOLDER

| ASSERT | Variable tested | Expected value | Description |
|-----------|------------------------------------------|----------------|-----------------------------------------|
| NOT EQUAL | documentFolder | undefined | Object was created and is not undefined |
| NOT EQUAL | documentFolder .getFiles() .length | 0 | Number of files is greater than 0 |

GET BY ID

| ASSERT | Variable tested | Expected value | Description |
|-----------|--------------------------------------------------------------------|----------------|------------------------------------------------|
| NOT EQUAL | documentFolder .getById("S1000DBIKE-AAA-D00-00-00-00AA-00WA-D") | undefined | The document with requested id is returned |
| EQUAL | documentFolder .getById("none") | undefined | Id without matching document returns undefined |

GET BY POINTER

| ASSERT | Variable tested | Expected value | Description |
|---------------|--------------------------------------------------------------------------------------------------|-----------------------|-----------------------------------------------------------|
| NOT EQUAL | documentFolder .getByPointer("DMC-S1000DBIKE- AAA-D00-00-00-00AA-330A-A_007- 00_EN-US") | undefined | The document with requested pointer is returned |
| EQUAL | documentFolder.getByPointer("none") | undefined | Pointer without matching document returns undefined |

DOCUMENT MODEL

SCENARIO

| | |
|--------------|---------------------------------------------------------------|
| GIVEN | A document from CORENA pinpoint neutral package exists |
| WHEN | A document model is created from the CORENA pinpoint document |
| THEN | The properties of the document model are set correctly |

UNIT TESTS

| | | |
|--------------------------------|-------------------------------------------------------------------|--------|
| Document Model Creation | The attributes of the document model is set to the correct values | Passes |
|--------------------------------|-------------------------------------------------------------------|--------|

ASSERTS

DOCUMENTMODEL CREATION

| ASSERT | Variable tested | Expected value | Description |
|--------|-----------------------------------|---------------------------------------------------------|-------------------------------------------------|
| EQUAL | documentModel_1 .getPointer() | "DMC-S1000DBIKE-AAA-D00-00-00-00AA-00QA-D_004-00_EN-US" | The pointer of the loaded document is correct |
| EQUAL | documentModel_1 .getTitle() | "Mountain bicycle - Conditions cross-reference table" | The title of the loaded document is correct |
| EQUAL | documentModel_1 .getId() | "S1000DBIKE-AAA-D00-00-00-00AA-00QA-D" | The id of the loaded document is correct |
| EQUAL | documentModel_1 .getFilterId() | "0" | The filter id of the loaded document is correct |

PUBLICATION MANAGER

SCENARIO

| | |
|--------------|-------------------------------------------------------|
| GIVEN | A publication folder exists |
| WHEN | The publication is created in the publication manager |
| THEN | A publication can be loaded by folder name |
| AND | A publication can be loaded by id |

UNIT TESTS

| | | |
|---------------------------|--------------------------------------------------------|---------------|
| Create publication | An instance of publication is created correctly | Passes |
| Load by name | The correct publication is loaded from folder name | Passes |
| Load by id | The correct publication is loaded from id | Passes |

ASSERTS

CREATE PUBLICATION

| ASSERT | Variable tested | Expected value | Description |
|--------|-----------------------|----------------------------------------|----------------------------------------------------------|
| EQUAL | publication.getName() | "S1000D_Bike_Set" | The title of the publication is set to the correct value |
| EQUAL | publication.getId() | "f9c66907-68af-4b45-8d72-50575cdf1aea" | The UUID of the publication is set to the correct value |

LOAD BY NAME

| ASSERT | Variable tested | Expected value | Description |
|--------|-------------------------------------|----------------------------------------|---------------------------------------------------|
| EQUAL | publication.getId() | "f9c66907-68af-4b45-8d72-50575cdf1aea" | The publication with the correct UUID is returned |
| EQUAL | manager .loadByName("undefined") | undefined | Returns undefined if publication doesn't exist |

LOAD BY ID

| ASSERT | Variable tested | Expected value | Description |
|---------------|------------------------|-----------------------|---------------------------------------------------------------------------------------------------|
| EQUAL | publication.getName() | "S1000D_Bike_Set" | "publication.getName(), \"S1000D_Bike_Set\"; equal fails" message is returned at test fails |
| EQUAL | publication | undefined | "publication, undefined; equal fails" message is returned at test fails |

PUBLICATION MODEL

SCENARIO

| | |
|--------------|------------------------------------------|
| GIVEN | A folder with a publication exist |
| WHEN | A publication model is created |
| THEN | The documents are available |
| AND | The current document filter is available |
| AND | The table of contents is available |

UNIT TESTS

| Unit Tests | | |
|----------------------|------------------------------------------------|--------|
| Creation | Publication Model is created correctly | Passes |
| Get Documents | Document folder is accessible from publication | Passes |
| Get Filter | Filter object is accessible from publication | Fails |
| Get Toc | Toc object is accessible from publication | Passes |

ASSERTS

CREATION

| ASSERT | Variable tested | Expected value | Description |
|----------|-----------------|------------------------|----------------------------------------|
| NOTEQUAL | folder.path | undefined | Publication is not instantiated at all |
| EQUAL | folder.path | publicationFolder.path | Publication not instantiated correctly |

GET DOCUMENTS

| ASSERT | Variable tested | Expected value | Description |
|---------------|----------------------------|-----------------------|----------------------------------------|
| NOTEQUAL | publication.getDocuments() | undefined | Document folder isn't loaded correctly |

GET FILTER

| ASSERT | Variable tested | Expected value | Description |
|---------------|-------------------------|-----------------------|----------------------------|
| NOTEQUAL | publication.getFilter() | undefined | Loaded filter is undefined |

GET TOC

| ASSERT | Variable tested | Expected value | Description |
|---------------|------------------------|-----------------------|-------------------------|
| NOTEQUAL | publication.getToc() | undefined | Loaded ToC is undefined |

APPLICABILITY

ASSERT

SCENARIO

| | |
|--------------|------------------------------------------------|
| GIVEN | A valid assert exists |
| WHEN | A valid statement is evaluated to the assert |
| THEN | The applicability of the statement is returned |

UNIT TESTS

| | | |
|-------------------------------------|--------------------------------------------------------------|---------------|
| Applicability filter | Only applicable filter statements evaluate as true | Passes |
| Partial applicability filter | Statements where the value is not specified evaluate as true | Passes |
| Value range | Only values within the range specified evaluate as true | Fails |
| Value selection | Only values contained in the selection evaluate as true | Fails |

ASSERTS

APPLICABILITY
FILTER

APPLICABILITY FILTER

| ASSERT | Variable tested | Expected value | Description |
|--------|----------------------------------------------|----------------|-------------------------------------------------------|
| EQUAL | assertTestObject.assert(validApplicFilter) | true | Check if filter is applicable, returns if matches |
| EQUAL | assertTestObject.assert(invalidApplicFilter) | false | Check if filter is applicable, returns if not matches |

PARTIAL APPLICABILITY FILTER

| ASSERT | Variable tested | Expected value | Description |
|---------------|--------------------------------------|-----------------------|-----------------------------------------------|
| EQUAL | assertTestObject.assert(versionOnly) | true | Check if some part/version matches the filter |
| EQUAL | assertTestObject.assert({}) | true | Check if there is no filter |

VALUE RANGE

| ASSERT | Variable tested | Expected value | Description |
|---------------|------------------------------------------------------|-----------------------|----------------------------------------------------|
| EQUAL | rangeAssertTestObject. .assert(valueRange) | true | Check if detected value is in the authorized range |
| EQUAL | rangeAssertTestObject. .assert(invalidValueRange) | false | Check if the value is not in the range |

VALUE SELECTION

| ASSERT | Variable tested | Expected value | Description |
|---------------|----------------------------------------------------------|-----------------------|------------------------------------|
| EQUAL | selectionAssertTestObject. .assert(valueRange) | true | Check the selected value is valid |
| EQUAL | selectionAssertTestObject. .assert(invalidValueRange) | false | Check if selected value is invalid |

LIST

SCENARIO

| | |
|--------------|--------------------------------------------------------|
| GIVEN | A valid applic list exists |
| AND | A valid, applicable filter statement exists |
| WHEN | The filter statement is evaluated with the applic list |
| THEN | True is returned |

| | |
|--------------|--------------------------------------------------------|
| GIVEN | A valid applic list exists |
| AND | A valid, unapplicable filter statement exists |
| WHEN | The filter statement is evaluated with the applic list |
| THEN | false is returned |

UNIT TESTS

| Unit Test | | |
|-------------------------|--------------------------------------------------------------------|--------|
| List creation | The list is created with the correct objects and evaluate function | Passes |
| Evaluate and | And function evaluate correctly | Passes |
| Evaluate or | Or function evaluates correctly | Passes |
| Evaluate sublist | Sublists evaluate correctly | Passes |

ASSERTS

| LIST CREATION | | | |
|---------------|-----------------|----------------|----------------------------------------------|
| ASSERT | Variable tested | Expected value | Description |
| EQUAL | hasObjects | true | Check if list-object is created successfully |

| EVALUATE AND | | | |
|-----------------|----------------------------------|-------------------|---------------------------------------------|
| ASSERT | Variable tested | Expected value | Description |
| EQUAL | andList.assert(mountainStormMK1) | true | Check if filter is matched and |
| EQUAL | andList.assert(modelOnly) | True | Check if filter is matched and |
| EQUAL | andList.assert(brookTrekkerMK1) | False | Check if filter is not matched and |
| EQUAL | andList.assert({}) | ture | Check if filter is empty and |

| EVALUATE OR | | | |
|----------------|---------------------------------|-------------------|-----------------------------------------------|
| ASSERT | Variable tested | Expected value | Description |
| EQUAL | orList.assert(mountainStormMK1) | true | Check if filter is matched or |
| EQUAL | orList.assert(modelOnly) | true | Check if filter is matched or |
| EQUAL | orList.assert(brookTrekkerMK1) | true | Check if filter is not matched aornd |
| EQUAL | orList.assert({}) | true | Check if filter is empty or |

EVALUATE SUBLIST

| ASSERT | Variable tested | Expected value | Description |
|---------------|---------------------------------------|-----------------------|----------------------------------------------------------------|
| EQUAL | SubLists .assert(mountainStormMK1) | true | Check if applicable filter (given parameter) matches "SUB" |
| EQUAL | subLists .assert(modelOnly) | True | Check if applicable filter (given parameter) matches "SUB" |
| EQUAL | subLists .assert(brookTrekkerMK1) | False | Check if applicable filter (given parameter) not matches "SUB" |
| EQUAL | subLists .assert({}) | true | Check if empty filter matches "SUB" |

STATEMENT

SCENARIO

| | |
|--------------|---------------------------------------------|
| GIVEN | Given a valid statement exists |
| AND | A valid, applicable filter statement exists |
| WHEN | The filter statement is evaluated |
| THEN | True is returned |

| | |
|--------------|-----------------------------------------------|
| GIVEN | Given a valid statement exists |
| AND | A valid, inapplicable filter statement exists |
| WHEN | The filter statement is evaluated |
| THEN | False is returned |

UNIT TESTS

| Unit Test | | |
|----------------------------------|----------------------------------------------------|--------|
| Statement object creation | The statement object is initialized correctly | Passes |
| Applicability assert | Only applicable filter statements evaluate as true | Passes |

ASSERTS

| STATEMENT OBJECT CREATION | | | |
|---------------------------------|---------------------------------------|-------------------|----------------------------------------------------------------|
| ASSERT | Variable tested | Expected value | Description |
| EQUAL | emptyStatement.applic | undefined | Verifies if empty statement has no applic |
| NOTEQUAL | statement.applic | undefined | Checks if statement object is created successfully |
| OK | statement.applic instanceof list.List | true | Checks if list object is created successfully |

| APPLICABILITY ASSERT | | | |
|-------------------------|-----------------------------------------|-------------------|----------------------------------------------------|
| ASSERT | Variable tested | Expected value | Description |
| EQUAL | emptyApplic.evaluate(validApplicFilter) | true | Checks if empty statement has a valid filter |
| EQUAL | emptyApplic.evaluate({}) | true | Checks if empty statement has an empty filter |
| EQUAL | applic.evaluate(validApplicFilter) | true | Checks if there is any applicable filter |
| EQUAL | applic.evaluate(modelOnly) | true | Checks if given parameter has an applicable filter |
| EQUAL | applic.evaluate(invalidApplicFilter) | false | Checks if there is an inapplicable filter |
| EQUAL | applic.evaluate({}) | true | Checks on empty |

FILTER

SCENARIO

| | |
|--------------|---------------------------------------------------|
| GIVEN | The filter service has a set of statements loaded |
| AND | A valid filter statement exists |
| WHEN | The filter statement is evaluated |
| THEN | The applicable filter ids are returned |

UNIT TESTS

Unit Test

| | | |
|----------------------------|-------------------------------------------------------------------------------------------|--------|
| Match to filter ids | Only the applicable filter IDs are returned All the applicable filter IDs are returned | Passes |
|----------------------------|-------------------------------------------------------------------------------------------|--------|

ASSERTS

| ASSERT | Variable tested | Expected value | Description |
|--------|-------------------------------|----------------|------------------------------------------|
| EQUAL | list1.length | 2 | Check if list_1 has correct size |
| EQUAL | list2.length | 2 | Check if list_2 has correct size |
| EQUAL | contains(list1, element) | true | Check if list_1 contains given parameter |
| EQUAL | contains(list2, element) | true | Check if list_2 contains given parameter |
| EQUAL | contains(list1, "822998450") | false | Check if list_1 contains "822998450" |
| EQUAL | contains(list2, "1410265599") | false | Check if list_2 contains "1410265599" |

BEACON

BEACON APPLICABILITY

SCENARIO

| | |
|--------------|-----------------------------------------------|
| GIVEN | Beacon data has been recieved |
| AND | The beacon is registered with the application |
| WHEN | The filter data is requested |
| THEN | Then the filter statement is returned |

UNIT TESTS

| Unit Test | | |
|----------------------------------|----------------------------------------------------|--------|
| Read Beacon Applicability | Registered beacons return a valid filter statement | Passes |

ASSERTS

| READ BEACON APPLIC | | | |
|--------------------------|---------------------------|-------------------------|-------------------------------------|
| ASSERT | Variable tested | Expected value | Description |
| EQUAL | JSON.stringify(statement) | JSON.stringify(beacons) | Checks if JSON is read successfully |

BEACON MANAGER

SCENARIO

| | |
|--------------|----------------------------------------------------|
| GIVEN | A beacon manager has been created |
| WHEN | The beacon manager receives a list of beacons |
| THEN | The beacons displayed is updated with the new list |
| AND | Beacons not received are not displayed |

UNIT TESTS

| Unit Test | | |
|--------------------------------------|-----------------------------------------------------------|--------|
| Beacon manager initialization | Manager is initialized and beacons can be added | Passes |
| Beacon manager update | Beacons out of range are removed from the list | Passes |
| Beacon manager proximity | Beacons are sorted by proximity (measured rssi strength). | Passes |

ASSERTS

| BEACON MANAGER INITIALIZATION | | | |
|----------------------------------------------|-------------------------------|---------------------------|--------------------------------------------------------|
| ASSERT | Variable tested | Expected value | Description |
| EQUAL | manager.size | 0 | Checks if initialization is empty |
| EQUAL | manager.size | 2 | Checks if after update, beacons are added successfully |
| OK | manager.contains(1675, 12907) | true | Checks if beacon 1 is added successfully |
| OK | manager.contains(60845, 9963) | true | Checks if beacon 2 is added successfully |
| NOTOK | manager.contains(1234, 12345) | false | Assure that unknown beacons are not present |

| BEACON MANAGER UPDATE | | | |
|--------------------------------------|-------------------------------|-----------------------|--------------------------------------------------------|
| ASSERT | Variable tested | Expected value | Description |
| EQUAL | manager.size | 1 | Checks if beacon list is updated |
| OK | manager.contains(1675, 12907) | true | Assure that list includes beacons in the defined range |
| NOTOK | manager.contains(60845, 9963) | false | Assure beacons leaves range is removed from the list |

| BEACON MANAGER PROXIMITY | | | |
|-----------------------------------------|---------------------------------------------|-----------------------|-----------------------------------------------------------|
| ASSERT | Variable tested | Expected value | Description |
| OK | manager.list[0].rssi < manager.list[1].rssi | true | Assure that list is sorted by proximity |
| OK | manager.list[0].rssi < manager.list[1].rssi | true | Assure that sorted list is updates after changes in range |

BEACON

SCENARIO

| | |
|--------------|--------------------------------------|
| GIVEN | An iBeacon package has been received |
| WHEN | A beacon object is created |
| THEN | The beacon id is available |
| AND | The beacon RSSI is available |

UNIT TESTS

Unit Test

| | | |
|----------------------|------------------------------------------------|--------|
| Create beacon | Beacon object is initialized with correct data | Passes |
|----------------------|------------------------------------------------|--------|

ASSERTS

CREATE
BEACON

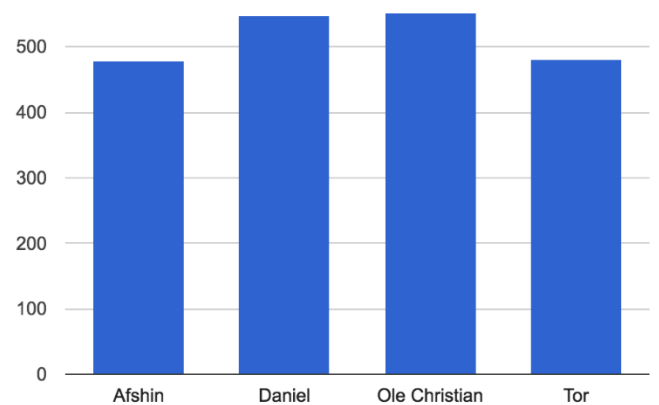
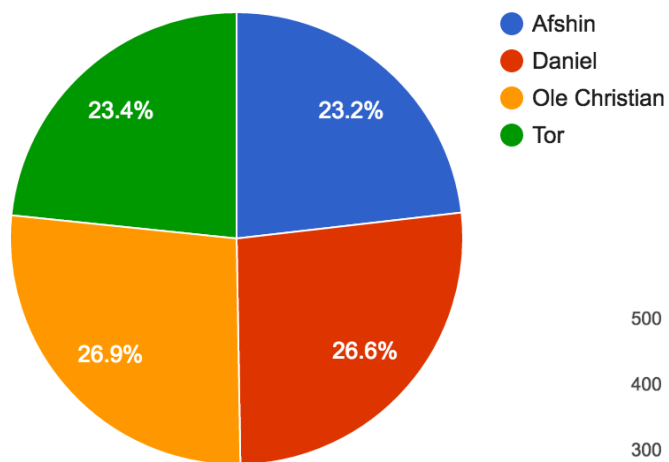
| ASSERT | Variable tested | Expected value | Description |
|--------|---------------------|----------------|--------------------------------------------------------|
| OK | beacon != undefined | true | Checks if beacon is created correctly and successfully |
| EQUAL | beacon.getId() | "1675-12907" | Checks if there is access by id |

APPENDIX E – BUDGET

| Description | Estimate | Approx | Comments |
|--------------------------|----------|--------|--------------------------------|
| Jira | 600 | 600 | |
| Beacons | 1000 | 1000 | |
| Printing 1. presentation | 100 | 0 | Printed at school |
| Printing 2. presentation | 100 | 0 | Printed at Flatirons Solutions |
| Printing 3. Presentation | 200 | | |
| Uniforms | 1600 | 1750 | Payed by students |
| Poster | 200 | | |
| Total | 3800 | | |

APPENDIX F – TIMESHEETS

| | Pre-sprints | Sprint 1 | Sprint 2 | Sprint 3 | Sprint 4 | Sprint 5 | Sprint 6 | Sprint 7 | Sprint 8 | |
|-----------------|-------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|----------|-------|
| Afshin | 2 | 40 | 52 | 26 | 27 | 30 | 23 | 36 | 39 | |
| Daniel | 5 | 35.5 | 60 | 27 | 21 | 32 | 28 | 37 | 55.5 | |
| Ole Christian | 5 | 35 | 59 | 23 | 29 | 32 | 30 | 37 | 55 | |
| Tor | 5 | 34 | 59 | 28 | 24 | 24.5 | 28 | 35 | 44.5 | |
| Sprint sum | 17 | 144.5 | 230 | 104 | 101 | 118.5 | 109 | 145 | 194 | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| Sprint duration | | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | |
| | | | | | | | | | | |
| | Sprint 9 | Sprint 10 | Sprint 11 | Sprint 12 | Sprint 13 | Sprint 14 | Sprint 15 | Sprint 16 | | Total |
| Afshin | 9 | 41 | 16 | 29 | 21 | 31 | 19 | 37 | | 478 |
| Daniel | 10 | 34 | 24 | 27 | 34 | 36 | 32 | 50 | | 548 |
| Ole Christian | 5 | 38 | 17 | 28 | 30 | 41 | 34 | 57 | | 555 |
| Tor | 11 | 27.5 | 7 | 30.5 | 21.5 | 31 | 20 | 51.5 | | 482 |
| Sprint sum | 17 | 144.5 | 230 | 104 | 101 | 118.5 | 109 | 145 | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| Sprint duration | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | |
| | Web Science | | Easter | | | | | | | |



APPENDIX G – TIMELINE

