Software Project Management 2015/2016

Smart shopping list

**Team Members:** Stefan Viktoler (team leader), Filip Michnik, Wang Yongjian, Udeshkumar Ganesan, Amélie Lagadec

**Project Proposal**

**Version 0.5**

History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Remarks |
| 0.1 | 24/11/2015 | Amélie Lagadec | Creation of document |
| 0.2 | 01/12/2015 | Amélie Lagadec | Strategic plan, Test plan, Schedule |
| 0.3 | 05/12/2015 | Wang Yongjian(Adam) | Problem statement |
| 0.4 | 07/12/2015 | Wang Yongjian(Adam) | Business case, Risk Management |
| 0.5 | 08/12/2015 | Filip Michnik | Executive overview, Communication plan, Legal issues |
|  |  |  |  |
|  |  |  |  |

Table of contents

[1 Executive overview 4](#_Toc437335549)

[1.1 The project 4](#_Toc437335550)

[1.2 Benefits and Forecast 4](#_Toc437335551)

[2 Problem statement 5](#_Toc437335552)

[2.1 Background 5](#_Toc437335553)

[2.2 Project mission 6](#_Toc437335554)

[3 Business Case 7](#_Toc437335555)

[3.1 Profits 7](#_Toc437335556)

[3.2 Safety 7](#_Toc437335557)

[3.3 Benefit to the community 8](#_Toc437335558)

[4 Risk Management 8](#_Toc437335559)

[4.1 IRE through Identify, Rank and Evaluate to mitigate risks 9](#_Toc437335560)

[4.2 Simple Risk Estimate/Breakeven Point based on Standish Report 10](#_Toc437335561)

[5 Cost Estimate 10](#_Toc437335562)

[6 Communication Plan 10](#_Toc437335563)

[6.1 Communication Plan 10](#_Toc437335564)

[6.2 Weekly Team Status Report 12](#_Toc437335565)

[6.3 Responsibility assignment matrix (RACI) 12](#_Toc437335566)

[7 Legal Issues 13](#_Toc437335567)

[7.1 Data Privacy Law 13](#_Toc437335568)

[7.2 Support and warranty mechanisms 14](#_Toc437335569)

[7.3 Proprietary rights 14](#_Toc437335570)

[8 Staffing Plan 15](#_Toc437335571)

[8.1 Participants in the project 15](#_Toc437335572)

[8.2 Briggs Meyers compatibility matrix 15](#_Toc437335573)

[9 Strategic Plan 16](#_Toc437335574)

[9.1 Balanced Scorecard 16](#_Toc437335575)

[9.2 SWOT 16](#_Toc437335576)

[10 Test Plan 17](#_Toc437335577)

[10.1 Functions to be tested 17](#_Toc437335578)

[10.2 Functions not to be tested 17](#_Toc437335579)

[10.3 Testing process overview 17](#_Toc437335580)

[10.4 Testing Strategy 19](#_Toc437335581)

[10.4.1 Data and Database Integrity Testing 19](#_Toc437335582)

[10.4.2 Function testing 19](#_Toc437335583)

[10.4.3 User interface testing 20](#_Toc437335584)

[10.4.4 Security and Access Control Testing 20](#_Toc437335585)

[10.4.5 Configuration and installation Testing 20](#_Toc437335586)

[11 Schedule 21](#_Toc437335587)

[12 Design structure matrix 23](#_Toc437335588)

# Executive overview

## The project

This report elicits key points of a software projects which purpose is to develop an application for Smart phones. This application allows its user to analyze data that is generated by a smart object (refrigerator) in an efficient and intuitive way. With potential uses cases such as nutrition assistance and easier household management (shopping lists, alerts on perishables goods, auto-shopping e.g.) it provides an adequate answer to current social trends such as the ephemeral nature of modern life, increasing digitalization, desire for efficiency and a growing health- and nutrition-consciousness.

[SOFTWARE MOCKUP]

Considering additionally current developments in science and technology this software provides an opportunity to enter into a new and potentially lucrative market.

## Benefits and Forecast

The GBI Research group published a study on the growth of the “smart” industry in 2013. As can be seen in Figure 1 the growth for Smart Home related business cases is expected to double by 2017.

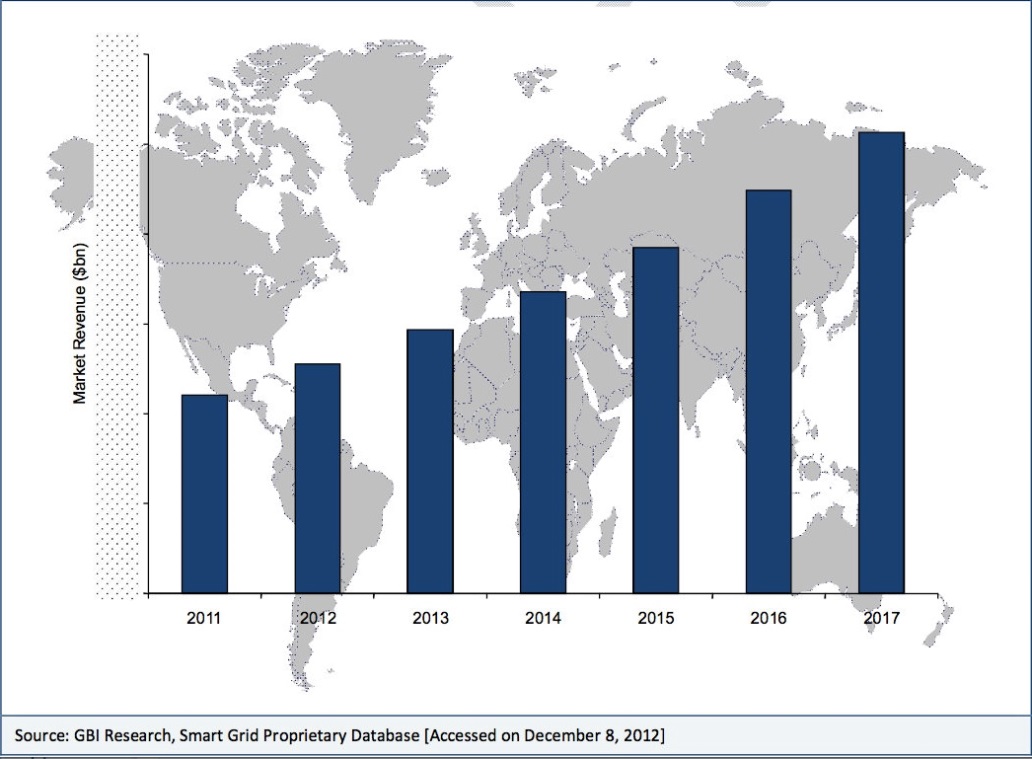


Abbildung 1 Smart Home Market Revenue 2011-17

In the meanwhile statista [1] published a similar prognosis, assuming that America’s Smart Home market will grow to 22 billion in 2020, confirming possible prospects of success. Gartner’s Hype Cycle is naming the Internet of Things as one of the current hypes that will reach its peak in 5 to 10 years. [2] Considering the fact that many of the world top leading economies are currently rolling-out government substituted Internet of Things programs in order to fund their manufacturing industries [3], it is highly advisable to deploy potential business cases early on.

Taking into account that developments like the Internet of Things or Smart technology will likely affect the business not only of refrigerator manufacturers, this software provides a chance to enter the Smart Home market with an innovative business case and enabling investors to benefit from early adopter clientele.

According to the theory of Diffusion of Innovations [4], early adopters provide producers with valuable feedback and are willing to be exposed to risks and problems, which allows producers to establish itself as early forces in new markets, hence having a knowledge leverage once business cases and revenue becomes more mature.

Since this software will be developed with a five person team and in an early stage of the Smart Home market, this projects spots relatively low costs (see section XY) compared to its benefits. While it is hard to give estimated figures on revenue, this software has a wide range of possible uses. The product management can consider marketing it as additional software to existing costumers of their brand of refrigerators. Thus, allowing a diversification of the product portfolio and ensuring positive product revenue and further incentives to buy their products.

From a client perspective and seeing that health application are generally a growing trend according to a recent PwC Study [5], the use cases stated above exhibit a high potential of consumer acceptance.

# Problem statement

## Background

The goal of the application allows its user to analyze data that is generated by a smart object (refrigerator) and depict the analyzation in an efficient and intuitive way. The application can be seen as a solution to the Smart Home environment which is part of the greater Internet of things development.

The Internet of things is basically a network of physical objects (so called “things”) that had been embedded with software and equipped with sensors and network connectivity, thus enabling them to collect and exchange data with each other. These objects can analyze their environment and can be controlled remotely across existing network structures such as the TCP/IP Protocol. Objects have a unique identifier (such as their IP-Address) and in cooperate themselves into the existing Internet infrastructure.

“Things” can refer to a wide range of physical objects, such as manufacturing plants, cars, pallets in a warehouse or refrigerators, amongst others. All these objects are endowed with specific embedded software and sensors (e.g. RFID-Readers) that can generate data based on their input data. Smart objects can feed their observations into the existing Internet, thus allowing objects or applications to exchange information with each other.

The smart home is the environment created by turning physical objects of households into smart objects. Possibilities are versatile from smart stoves and pots, which automatically alert their owners, to smart refrigerators, with groceries that are equipped with RFID-Tags, allowing users to analyze their nutrition habits, inform them of shortages or generate shopping lists. The varieties of possible application are huge.

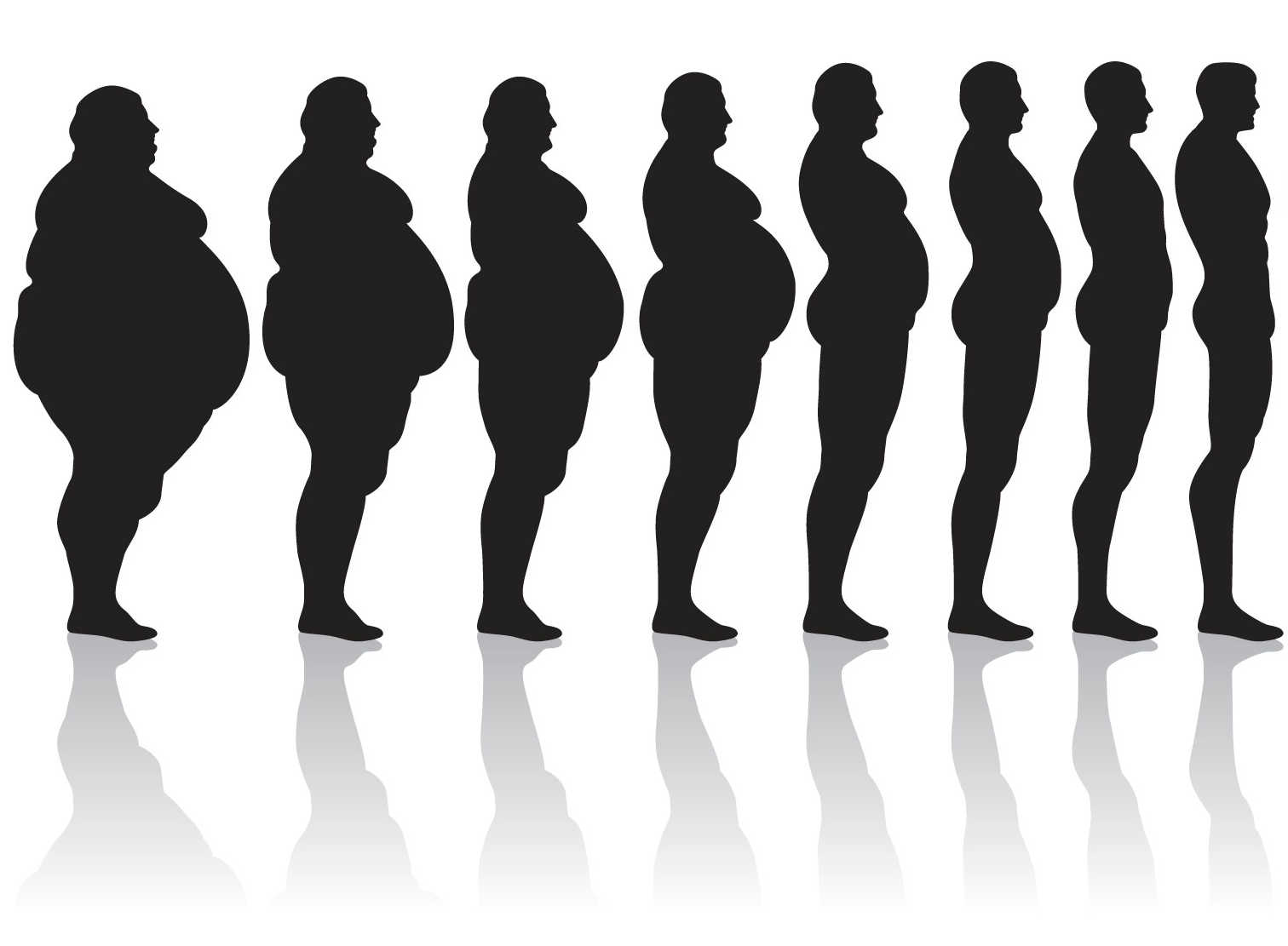
## Project mission

Regarding a full refrigerator like below, sometimes we could forget the food until it becomes spoiled. It happens and happens a lot. What a waste! We would like to solve the problem through an application.



A full refrigerator

Preventing the food from spoiling is not enough, what’s more? Fat population is bigger and bigger. Exercise is important and a balance diet is also important, we would like everyone to have a good health. We could achieve this by our smart shopping list application.



Fat can be thin

Problem 1: People left food in their refrigerator until it become spoiled.

Problem 2: The fat population becomes bigger and bigger, people need a balanced diet.

The purpose of this project is to develop a mobile application which supports different mobile operating systems and analyzes the data generated by a smart refrigerator and reminds the user in case that the user may forget the food which is going to spoil. Generating a balance and health shopping list will save you a lot of time and gives you a healthier life.

# Business Case

## Profits

As smartphone and internet of things developing, our life has changed a lot. It is a trend that our life will be more intelligent and those companies which seize this opportunity will have a great business.

For the refrigerator company, selling intelligent refrigerator will make a great profit and make a revolution of the traditional refrigerator.

For the customers, investing on health is really worthy. They can now buy a smart refrigerator, prevent food from spoiling and have a healthy diet.

## Safety

The product is not a strange new thing that hard for customer to accept. The internet of things has developed itself for many years and lots of matured products have appeared in the market. We already live in a life surrounded by a lot of smartphone applications. We use applications to manager our bank account, to listen to music, to watch video, to communicate with friends, and even to do business.

Regarding the investment, it won’t be huge and the risks of the product are small. We are not going to develop a fresh new refrigerator, we are enhancing the current refrigerator hardware with sensors and mobile application. It is not as hard as invent a new kind of refrigerator but the benefit will be the same as developing a new refrigerator.

## Benefit to the community

The smart shopping list benefits the community of consumers in which every people belongs. Indeed, we don't have to create a shopping list anymore before heading out to the supermarket. The users will not have to change any behavior to maintain this inventory list and thanks to the application we never eat stale or spoiled food. Over time, we will be able to see the patterns of our food consumption and better manage our household budget.

# Risk Management

Every aspect of software project management involves some form of risk management. For our project there are several possible risks.

**Risk 1:** We are software engineers not familiar with the hardware

**Plan to mitigate:** Ask the refrigerator company RD members to train us.

**Risk 2**: The 2 main mobile platforms development are quite different, parallelly develop two different platforms mobile app will cost a lot of manpower.

**Plan to mitigate:** Focus on iOS platform first, after iOS app finished we begin the Android app development.

**Risk 3:** we don't have QA team, software quality may not meet the requirements.

**Plan to mitigate:** 1. Several phases test, Development Test, Alpha Test, Beta Test

2. Outsource the test.

**Risk 4:** if too many new features needed to be added and the requirements changes out of control.

**Risk 5**: if there is no enough time budget to adapt the software to so many different android phones.

The most effective means of managing risk is by using a combination of strategic (that is, planning to avoid risk) and tactical (that is, preparing to take remedial action if adverse events occur) methods. Here we will use two risk mitigation methods to manage the risks.

## IRE through Identify, Rank and Evaluate to mitigate risks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Relative Severity** | **Relative Probability** | **Phase** | **Description** | **Mitigation** |
| 1 | 1 | Design | Not familiar with the hardware | Ask the refrigerator company RD members to train us. |
| 3 | 2 | Coding | Two main mobile platforms development are quite different, parallelly develop two different platforms mobile app will cost a lot of manpower | iOS platform first, later Android app |
| 2 | 3 | Testing | Don't have QA team | 1.Several phases test, Development Test, Alpha Test, Beta Test 2.Outsource the test |
| 5 | 4 | Requirements acquirement | Requirements changes out of control. | freeze the requirements after achieving the requirements acquisition |
| 4 | 5 | Testing | No enough time budget to adapt the software to so many different android phones | 1.Strict API level to Android v2.3 2.Online Android compatibility test suit |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Relative Severity** | **Relative Probability** | **Phase** | **(Relative) Severity X Probability =** | **Relative Ranking based on the equation Severity X Probability** |
| 1 | 1 | Design | 1 | 1 |
| 3 | 2 | Coding | 6 | 2 |
| 2 | 3 | Testing | 6 | 3 |
| 5 | 4 | Requirements acquirement | 20 | 5 |
| 4 | 5 | Testing | 20 | 4 |

Based on above IRE table, we will focus on the top 3 risks. We will use each mitigation way to mitigate corresponding risk. And we will still monitor the other risks and new occurring risks.

## Simple Risk Estimate/Breakeven Point based on Standish Report

• Pf = probability a project is a type 2 or type 3 or ~ 71%

• R0 = the ratio of these projects’ overrun to the original budget to be ~ 56%

• Rf = the ratio of the cost of these projects to the original or ~ 156%

• B is the initial budget of the software project

Ee = PfBRf + (1-Pf)B (formula 1)

B = Ee/(PfR0 + 1) (formula 2)

RJ as a fraction of B.

Spending RJB on risk management activities reduces the average probability of failure Pf by a fraction, ∆Pf.

∆Pf is in units of the probability of failure

Ee = RJB + (Pf - ∆Pf)BRf + (1-(Pf - ∆Pf))B (formula 3)

∆Pf = RJ/RO (formula 4) [7] [8] [9]

Because we don’t have our history data, we will use the data from Chaos report. Here we will set aside budget for the risk management activity.

According formula 2 and data from chaos report, B = 1.7Ee, Rj=0.39. We will prepare additional 0.39B to mitigate the risks.

Finally we must keep in mind that no matter how greatly increase our chances of mitigating the impact of adverse events, there are no guarantees that we will avoid them altogether. So we will always monitor the risks during the whole lifecycle of the development.

# Cost Estimate

# Communication Plan

## Communication Plan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Communica-tion Type | Objective | Owner | Audience | Frequency / Timing | Documentation | Media |
| Introduction to Project | Provide Introduction, Identification of relevant Stakeholders, Finalization of schedule | Project Manager | Senior Management, Stakeholders | Once (at the Beginning) | PowerPoint  Word-Document (Report) | In-Person |
| Introduction to Project | Provide Introduction, Workshop appointments | Project Manager & Team | Project Team | Once (at the Beginning | PowerPoint | In-Person |
| Stakeholder Workshops / Interviews | Requirements Analysis, Design Specification, Risk Analysis, Appointment of Key Users | Project Team | Stakeholders, Users | To be defined | Word-Document | In-Person |
| Project Status Meeting | Update Work plan, Issue Resolution, Project Status | Project Manager & Team | Project Team | Bi-Weekly | Word-Document (Report) | In-Person |
| Ongoing Project Status | Show current Project Status/ schedule | Project Manager | Project Team | Weekly | Newsletter | E-Mail |
| Stakeholder Project Status Meeting | Evaluation of Project Progress and Expectations | Project Manager | Stakeholders,  Senior Management | To be determined in the introduction Meeting | Word-  Document  (Report) PowerPoint | In-Person |
| Formal Project Status | Tracking of Project Progress | Project Manager | Senior Management, Stakeholders | 3-Weekly | Report | E-Mail |
| Key User Workshop | Black Box Testing, Evaluation | Project Team | Stakeholders, Key Users | To be determined in Workshops | Word-Document (Report) | In-Person |
| Lessons Learned | Evaluation | Project Manager | Project Team | At the End of important Project Phases | Word-Document  Knowledge Database | In-Person |
| Project Handover | Project Commission | Project Manager | Senior Management, Stakeholders | Once (at the End) | Completion Certificate  Report | In-Person |

## Weekly Team Status Report

The weekly Project Status Report template is stated below. This communication procedure informs the team members and helps to maintain track of work progress. Team members are supposed to update this report according to their progress and clarify matters during the Bi-weekly Project Status meetings. A current summary is send to each member via E-Mail weekly.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Monthly Status Report (MM/YY)** | | | | |
| Completed | | | | |
| Code | Task | Team Member | Date Completed | Notes |
| Phase 1 | Stakeholder Interviews | X, Y, Z | MM/DD/YY | - |
| In Progress | | | | |
| Code | Task | Team Member | Estimated Completion Date | Notes |
| Phase 2 | Software Product Design | X,Y | MM/DD/YY | Waiting for Stakeholder input |
| Assigned But Not Started | | | | |
| Code | Task | Assigned Team Member | Estimated Start Date | Note |
| Phase 3 | Coding | X,Z | MM/DD/YY | Waiting for Completion of Phase 2 |

## Responsibility assignment matrix (RACI)

Roles: Responsible **(R)**, Accountable **(A)**, Consulted **(C)**, Informed **(I)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Code | Name | Project Manager | Developer | Designer | Tester |
| Phase 1 | Requirement | A | C | R | - |
| Phase 2 | Analysis | A | - | R | - |
| Phase 3 | Design | A | C | R | - |
| Phase 4 | Development | A | R | C | - |
| Phase 5 | Testing | A | C | A | R |
| Phase 6 | Deployment | A | C | R | C |

# Legal Issues

## Data Privacy Law

There are several legal provisions in information privacy and data protection laws that can affect the realization of this project.

Firstly, it must be noted that the current legal situation of data protection and information privacy is not uniform worldwide. Therefore, it is necessary to distinguish between countries or politico-economic unions (such as the European Union) individually. This section explores the obstacles on the basis of the EU’s information privacy laws, which are some of the most heavily regulated and actively enforced data privacy laws worldwide. [BEWEIS?]

According to the EU’s Data Protection Directive all “entities or persons which collect or process personal data”[ZITAT1] are “data controllers”[ZITAT] and thus have to follow the following four principles: [ZITAT]

* collect and process personal data only when this is legally permitted
* respect certain obligations regarding the processing of personal data
* respond to complaints regarding breaches of data protection rules
* collaborate with national data protection supervisory authorities

Under the terms of the Data Protection Directive personal data is considered to be “personal information such as [..] names, photographs, telephone numbers, birth date and address while engaged in a whole range of everyday activities.” [ZITAT]

Data records such as logs generated by the household’s refrigerator might be considered as personal data and as such have to be treated adequately. That is to say, that a contractual legal framework has to be provided that enables the company to operate as a legal “data controller”. Clients must be required to sign policies that allow the company to collect this type of data for their own benefits. In the future (assuming a growth in data sets) this may lead to the necessity of external data security audits and/or the creation of internal compliance control entities in order to operate within current law. It is advised to consult legal and technical specialist in advance to make sure that the hardware’s manufacturer has acquired all necessary rights and expectancies.

In Order to conform to those legal obligations, extra costs such as those for a Chef Privacy Officer, secure data storage and further additional administrative expenses must be taken into account. These costs are hard to estimate and vary obviously, depending on the size of the operation and strictness of legal provisions. Nevertheless, the monetary or public damage due to data breaches or user complaints can be fierce. [LINK PAPER1] According to an IBM Study on data breach the average cost of each lost or stolen record is about $154 in 2015[LINK2]

[ZITAT1] <http://ec.europa.eu/justice/data-protection/data-collection/index_en.htm>

[LINK PAPER1]

[LINK2] <http://www-03.ibm.com/security/data-breach/>

<http://ec.europa.eu/justice/data-protection/data-collection/index_en.htm>

## Support and warranty mechanisms

A formal 3-month warranty is issued, commencing after user-acceptance testing has completed. During this period all defects are fixed at no cost to the customer. Additional feature requests are carried out on a fee-for-service basis.

During the Requirement Phase a team will be created that provides user support for questions concerning hosted services. This systems administration team works with the customer to configure the initial setup of their service and program interfaces. This work includes consulting on the appropriate hosting arrangements if necessary.

After commissioning issues are triaged into support and defect cases. Defects are escalated to the engineering team. Responsibility for each support issue is assigned to a case manager, who looks after the reporter until the issue is either resolved or escalated.

Scheduled outages are negotiated with the customer as they are required e.g. for upgrade of hardware or maintenance releases of software.

Training is supported though appointed Key-Users and written material (e.g. Documentations)

## Proprietary rights

The Parties acknowledge that this is a Work for Hire and agree that the Company will hold all intellectual property rights in the Project including, but not limited to, copyright and trademark rights in any deliverables pertaining to the Project. The Contractor agrees not to claim any such ownership in such intellectual property at any time prior to or after the completion of the Project.

# Staffing Plan

## Participants in the project

We work in a software company where basic infrastructure and developing techniques are available for this project. In this case we do not need to spend a lot of time thinking about best ways of storing data, communicating via TCP/IP and go through design principles.

The team is described as follow:

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | **Substitute** |
| **Stefan Viktoler** | Team leader | Filip Michnik, Amélie Lagadec |
| **Filip Michnik** | Developer / Designer | Wang Yongjian |
| **Wang Yongjian** | Developer / Designer | Filip Michnik |
| **Udeshkumar Ganesan** | Developer / Tester backup | Amélie Lagadec |
| **Amélie Lagadec** | Developer / Tester | Wang Yongjian |

In case of some problems, the team can be extended.

## Briggs Meyers compatibility matrix

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Stefan ENTP** | **Filip ENFP** | **Wang ESFJ** | **Udesh ESTP** | **Amélie ISFJ** |
| **Amélie ISFJ** | 0.50 | 0.33 | 0.50 | 0.33 |  |
| **Udesh ESTP** | 0.83 | 1.00 | 0.50 |  |  |
| **Wang ESFJ** | 0.67 | 0.50 |  |  |  |
| **Filip ENFP** | 0.83 |  |  |  |  |
| **Stefan ENTP** |  |  |  |  |  |

# Strategic Plan

## Balanced Scorecard

The following balanced scorecard is a strategic planning and management system that is used to align business activities to the vision and strategy of our organization, improve internal and external communications and monitor organization performance against strategic goals.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Objective** | **Measurement** | **Target** | **Initiative** | **Owner** | **Supplier** |
| **Financial** | -Lower Support costs  -increasing longterm revenue  - maximize returns | -Support resource costs  -Revenue  - ROCE | < 20% of the  software costs  -5% growth  - 14% | ITIL  restructuring | IT Department | Service Desk  employees |
| **Customer** | -User satisfaction  -Fast response time | -Customer Market survey  -flow time | -#1  -<48 hours | -Quality  management  -Problem  manager | -QM  Department  -IT Department | -Quality manager  -Service Desk  employees |
| **Internal** | - Better customer  knowledge  -Develop innovative projects  - Max return on ressource allocation | -Customer  service  - % R&D project  - employee productivity improvement | - 10% growth of market  share  - 80%  -5% | -Product  evolvement  - Project manager | -Release and  Marketing manager  - Project manager | - Company members  -Employees |
| **Learning** | - Help Desk Team  education  -Ensure market driven skills  - Leading employee satisfaction | - % of team  trained on  the time  - strategic skill coverage ratio  - Employee satisfaction rating (5 point scale) | - 100% in 1  year  -65%  - 4 | - Consultant  from Development  team  - Consultants  - Project manager | - Dual Role  Help Desk  and  Development  team  -Development team | - Help Desk and Development  team  -Development team |

## SWOT

The following SWOT matrix has the aim to evaluate the strengths, weaknesses, opportunities and threats involved in our business venture.

|  |  |  |
| --- | --- | --- |
|  | **Positive** | **Negative** |
| **Internal** | * Innovative aspects (smart shpping list) * Experience * Knowledge * Data * Quality * Location and geography * Price, value * Financial reserves | * Continuity * Support robustness * Lack of competitive strength * Cash flow * Our vulnerabilities * Reliability of data * Commitment |
| **External** | * Competitors vulnerability * New partnerships * New markets * New technologies * Fashion influences * Technology development and innovation * Customer satisfaction | * Competitor intentions * Economy * Legislative effects * Shift in consumer tastes * Substainable financial backing * New regulations |

# Test Plan

The Test Plan is designed to prescribe the scope, approach, resources, and schedule of all testing activities of the Smart shopping list project.

The plan identifies the items to be tested, the features to be tested, the types of testing to be performed, the personnel responsible for testing, the resources and the schedule required to complete testing. [10] [11]

## Functions to be tested

1. Graphical User Interface

2. Reports Output/Data

3. Report Setup/Locations

## Functions not to be tested

Not other than mentioned above in section 10.1.

## Testing process overview

**Understanding Requirements**

Requirement specifications will be sent by the client.

Understanding of requirements will be done by the whole team.

Raised queries will be sent by lead to the client.

Response to queries will be sent by the client.

**Preparing Test Cases**

Amélie Lagadec will be preparing test cases based on the requirement specifications. This will cover all scenarios for requirements. Also, the performance of the application for its responsiveness, scalability, resource usage and stability will be tested.

The usability testing will be taken in count (usability issues, navigation and content) as well as the security and compliance.

**Preparing Test Matrix**

Amélie Lagadec will be preparing test matrix which maps test cases to respective requirement. This will ensure the coverage for requirements.

**Reviewing test cases and matrix**

Peer review will be conducted for test cases and test matrix by Stefan Viktoler

In certain cases for e.g. complex requirements, lead's help will be taken for conducting review

Any comments or suggestions on test cases and test coverage will be provided by Udeshkumar Ganesan

Re-worked improvements will be reviewed and approved by Stefan Viktoler

**Creating Test Data**

Test data will be created by Amélie Lagadec on client's developments/test device based on scenarios and Test cases.

**Executing Test Cases**

Test cases will be executed by Amélie Lagadec on client's development/test device based on designed scenarios, test cases and Test data.

Test result (Actual Result, Pass/Fail) will updated in test case document

**Retesting and Regression Testing**

Retesting for fixed bugs will be done by Amélie Lagadec once it is resolved by Filip Michnik and bug/defect status will be updated accordingly. In certain cases, regression testing will be done if required.

**Deployment/Delivery**

Once all bugs/defect reported after complete testing is fixed and no other bugs are found, report will be deployed to client’s test site by the developers.

Once round of testing will be done by Amélie Lagadec on client’s test site a report will be delivered along with sample output by email to Stefan Viktoler.

After reviewing, Stefan Viktoler will send the report delivery email to the client.

## Testing Strategy

We can already provide the following testing strategy. It will be reviewed to add the client’s requirements.

### Data and Database Integrity Testing

The databases and the database processes should be tested as separate systems. These systems should be tested without the applications (as the interface to the data).

We will ensure Database access methods and processes function properly and without data corruption. This will be done by invoking each database access method and process, seeding each with valid and invalid data (or requests for data). We will inspect the database to ensure the data has been populated as intended, all database events occurred properly, or review the returned data to ensure that the correct data was retrieved (for the correct reasons).

Completion Criteria : All database access methods and processes function as designed and without any data corruption.

### Function testing

The goals of these tests are to verify proper data acceptance, processing, and retrieval, and the appropriate implementation of the business rules. We will thus verify the application by interacting with the application via the GUI and analyzing the output (results).

We will ensure proper application navigation, data entry, processing, and retrieval by executing each use case or function, using valid and invalid data. This will verify that the expected results occur when valid data is used, the appropriate error / warning messages are displayed when invalid data is used and that each business rule is properly applied.

Completion Criteria: All planned tests have been executed and all identified defects have been addressed.

### User interface testing

The goal of User Interface Testing is to ensure that the User Interface provides the user with the appropriate access and navigation through the functions of the applications. In addition, UI Testing ensures that the objects within the UI function as expected and conform to corporate or industry standards.

We will verify that navigation through the application properly reflects business functions and requirements, including window to window, field to field, and use of access methods. The window objects and characteristics, such as menus, size, position or state will be conformed to standards.

We will create / modify tests for each window to verify proper navigation and object states for each application window and objects.

Completion Criteria: Each window successfully verified to remain consistent with benchmark version or within acceptable standard.

### Security and Access Control Testing

We will verify that the user can access only those functions / data for which their user type is provided permissions and that only those users with access to the application are permitted to access them.

For that we will list each user type and the functions / data each type has permissions for. We will create tests for each user type and verify permission by creating transactions specific to each user type. We will modify user type and re-run tests for same users. In each case, we will verify that those additional functions / data are correctly available or denied.

Completion Criteria: For each known user type the appropriate function / data are available and all transactions function as expected and run in prior Application Function tests.

### Configuration and installation Testing

We will validate and verify that the client application is installed and function properly on the prescribed client mobile device.

# Schedule

To deliver a working application to display all required information (e.g. shopping list) which is connected via TCP/IP to the smart object at home, we assume that there is already an interface implemented to read the data given from the object. Therefore the main effort is in planning developing an intuitive design for the customer and user of the (shopping list) app.

The project will start the 4th of January 2016.

To achieve our goal we have to specify all requirements. This planning phase is estimated by 16 days, including all meetings with the refrigerator manufacturers, getting all information about the API to communicate, all needed functionalities for the app, the analysis of requirements and the documentation.

Thereafter we can start to design and prototype our application regarding to the specified requirements. For this task we calculate a time period of 40 days.

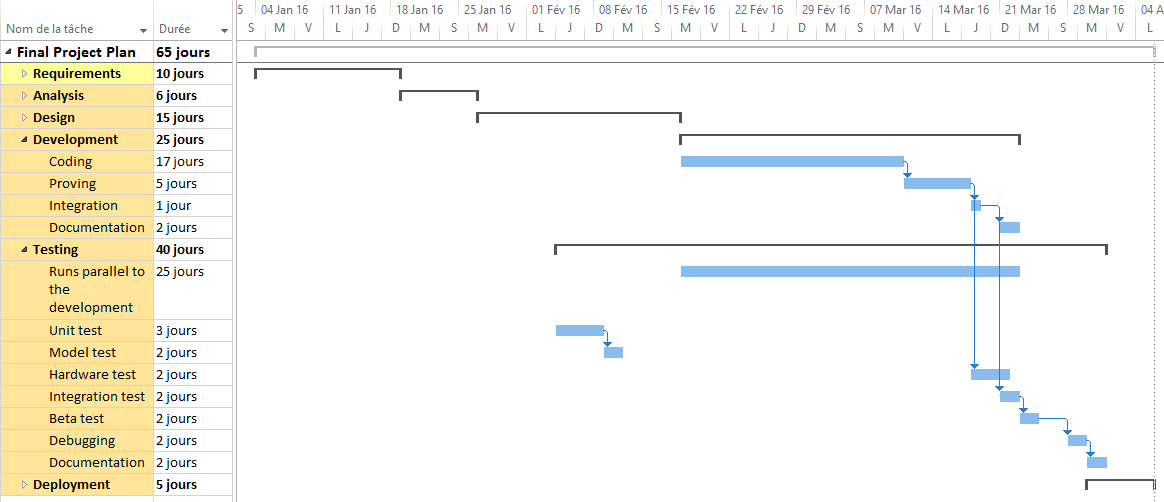
After finishing our first full functional app, we start to improve all designs and functionality, perform unit testing and acceptance testing until our final prototype and first productive application can be launched. This challenge is estimated by 40 days.

Then the project evolves to the support to maintain and possibly extend the application.

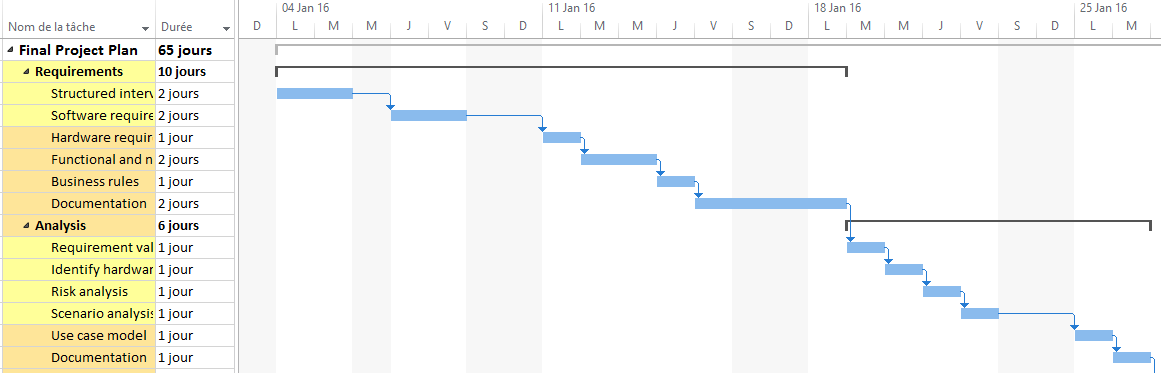
In total the development of an application suited for a smart home object requires 65 days.

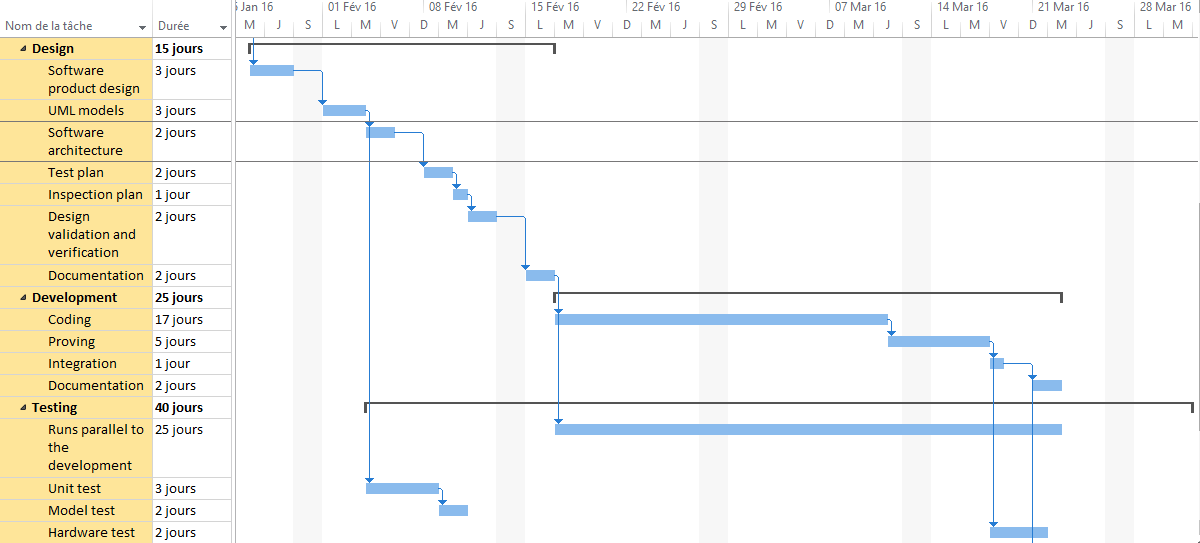
The project will end up on the 5th of April 2016.

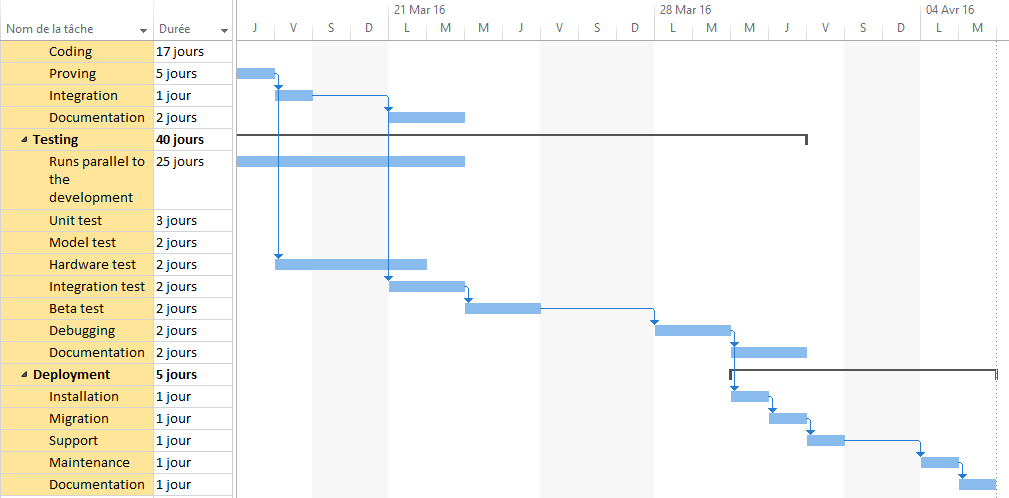
Here is the overview of the project in a Gantt chart format :



Here is the detailed different steps of the project :







# Design structure matrix

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**References**

[1] http://www.statista.com/statistics/296082/the-americas-smart-home-market-revenue/

[2] http://www.gartner.com/newsroom/id/3114217

[3] see the U.S.’s Manufacturing/AMP 2.0 program or Germany’s Industrie 4.0 for further reference. Just to name two examples.

[4] http://cleantechnica.com/2013/04/15/smart-cities/

[5] https://en.wikipedia.org/wiki/Early\_adopter

[6] http://www.pwc.com/us/en/health-industries/top-health-industry-issues.html

[7] Getting Results from Software Development Teams – Chapter 4: Developing and Maintaining the Project Plan: Section: Risk Managment - Microsoft Press 2008 Lawrence J. Peters

[8] A Simple Estimate of the Cost of Software Project Failures and the Breakeven

[9] Effectiveness of Project Risk Management - Stephen P. Masticola

[10] Course Registration System, Test Plan for the Architectural Prototype http://sce.uhcl.edu/helm/RUP\_course\_example/courseregistrationproject/artifacts/test/plans/test\_plan\_arch.htm#\_Toc449511176

[11] How to Create a Test Plan : http://www.guru99.com/what-everybody-ought-to-know-about-test-planing.html