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 0.8**

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, Background, Communication plan, Legal issues |
| 0.6 | 09/12/2015 | Amélie Lagadec | Design structure matrix, Staffing plan |
| 0.7 | 10/12/2015 | Stefan Vikoler | Cost estimation |
| 0.8 | 10/12/2015 | Filip Michnik | Format, Text |

Table of contents

1 Executive overview 5

1.1 The project 5

1.2 Benefits and Forecast 5

2 Problem statement 7

2.1 Background 7

2.2 Project mission 7

3 Business Case 9

3.1 Profits 9

3.2 Safety 9

3.3 Benefit to the community 9

4 Risk Management 10

4.1 IRE through Identify, Rank and Evaluate to mitigate risks 10

4.2 Simple Risk Estimate/Breakeven Point based on Standish Report 11

5 Cost Estimate 13

5.1 Labor cost 13

5.2 Overhead Expenses 14

5.3 Project Cost 15

5.4 Microsoft Project 15

5.5 Function Point Analysis with COCOMO II 17

5.6 Earned Value Management 21

6 Communication Plan 22

6.1 Communication Plan 22

6.2 Weekly Team Status Report 23

6.3 Responsibility assignment matrix (RACI) 24

7 Legal Issues 25

7.1 Data Privacy Laws 25

7.2 Support and warranty mechanisms 26

7.3 Proprietary rights 26

8 Staffing Plan 27

8.1 Participants 27

8.2 Briggs Meyers compatibility matrix 27

9 Strategic Plan 28

9.1 Balanced Scorecard 28

9.2 SWOT 28

10 Test Plan 30

10.1 Functions to be tested 30

10.2 Functions not to be tested 30

10.3 Testing process overview 30

10.3.1 Understanding Requirements 30

10.3.2 Preparing Test Cases 30

10.3.3 Preparing Test Matrix 31

10.3.3.1 Reviewing test cases and matrix 31

10.3.3.2 Creating Test Data 31

10.3.3.3 Executing Test Cases 31

10.3.3.4 Retesting and Regression Testing 31

10.3.3.5 Deployment/Delivery 31

10.4 Testing Strategy 32

10.4.1 Data and Database Integrity Testing 32

10.4.2 Function testing 32

10.4.3 User interface testing 32

10.4.4 Security and Access Control Testing 33

10.4.5 Configuration and installation Testing 33

11 Schedule 34

12 Design structure matrix 37

13 References 38

# 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. Potential use cases such as nutrition assistance and easier household management (shopping lists, recipes, alerts on perishables goods, auto-shopping amongst other things) provide a consequential answer to current social trends, such as the ephemeral nature of modern life, increasing digitalization, desire for efficiency and a growing consciousness of health- and nutrition related issues.

The envisioned software will be a smartphone application running on all current operating systems, with an emphasis on usability and intuitive presentation. Figure 1 depicts LG Electronics benchmark product and serves as a blueprint for this software. However; this project follows a stronger enhancement of health- and nutrition related use cases. Since health application are separately a growing trend[[1]](#footnote-1) the use cases stated above exhibit a high potential of consumer acceptance.



Figure 1 LG Smart Manager

## Benefits and Forecast

Considering current developments in science and technology, this project yields opportunity for entering into new and lucrative markets.

In 2013 the GBI Research Group published a study on “smart”-industry growth, assuming that Smart Home related business cases are expected to double by 2017. This development can be seen in Figure 2.

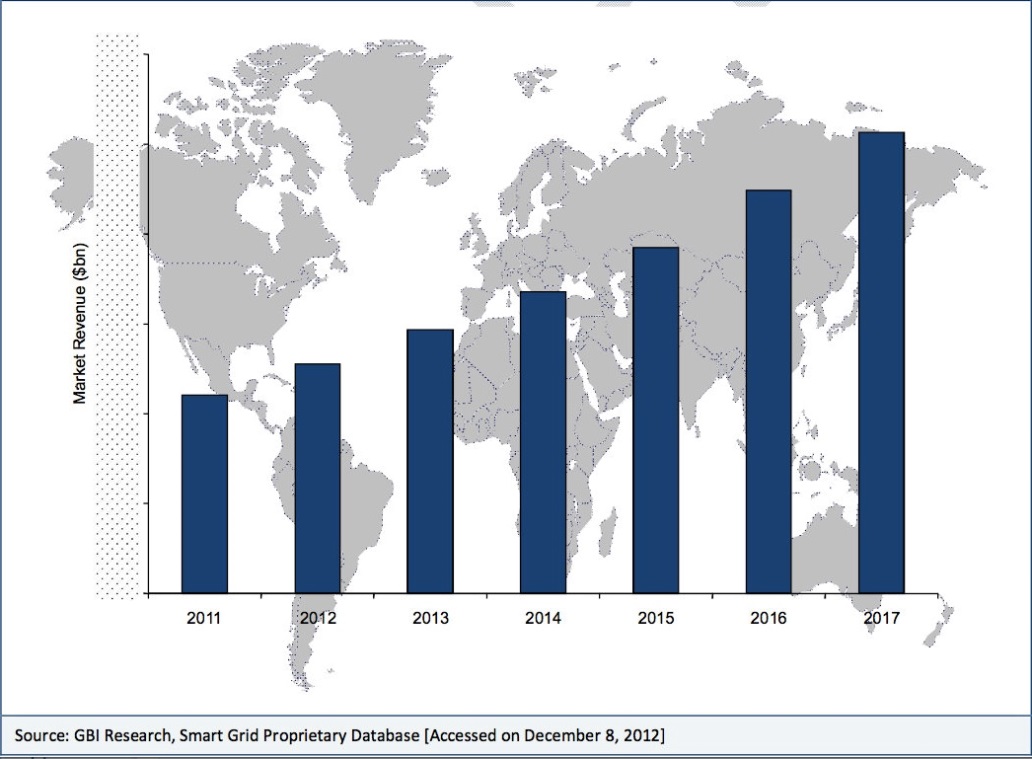


Figure 2 GBI Research Growth Revenue SM

A Statista [1] study, published that same year, confirms this prognosis, assuming that America’s Smart Home market will grow up to 22 billion in 2020.

Further, taking into account that Gartner’s Hype Cycle 2015 is naming the Internet of Things in general as one current trend, which will peak in 5 to 10 years [2] , it is advisable to watch all related market developments now. Many of the world’s top leading economies are already rolling-out government programs related to Smart Manufacturing. [3]

With respect to all this developments the Internet of Things or “Smart” trends are likely to affect future business models. This software provides a chance to respond to this with an innovative solution and enabling investors to benefit from early adopter clientele.[[2]](#footnote-2)

This software will be developed with a five person team and with relatively low costs (see section ) compared to its benefits. While it is hard to give estimated figures on revenue, the above mentioned use cases in cooperation with stated trends should serve as a proof that this project can bring benefit in terms of market share and innovation.

This application allows to be marketed as included in service or a separate product. Thus, allowing diversification of the product portfolio and in the long run ensuring positive product revenue and further incentives to buy other products.

# Problem statement

## Background

The goal of this 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

Coming from a user perspective and assuming a full refrigerator, as can be seen in Figure 3 Full refrigerator, spoiled food or even an unserviceable food stock (grown over time) occur frequently. This type of wasteful behavior can be solved through this application.



Figure 3 Full refrigerator

Besides waste prevention this application also intends to influence groceries selection in terms of health and nutrition, with the help of a smart shopping list feature.

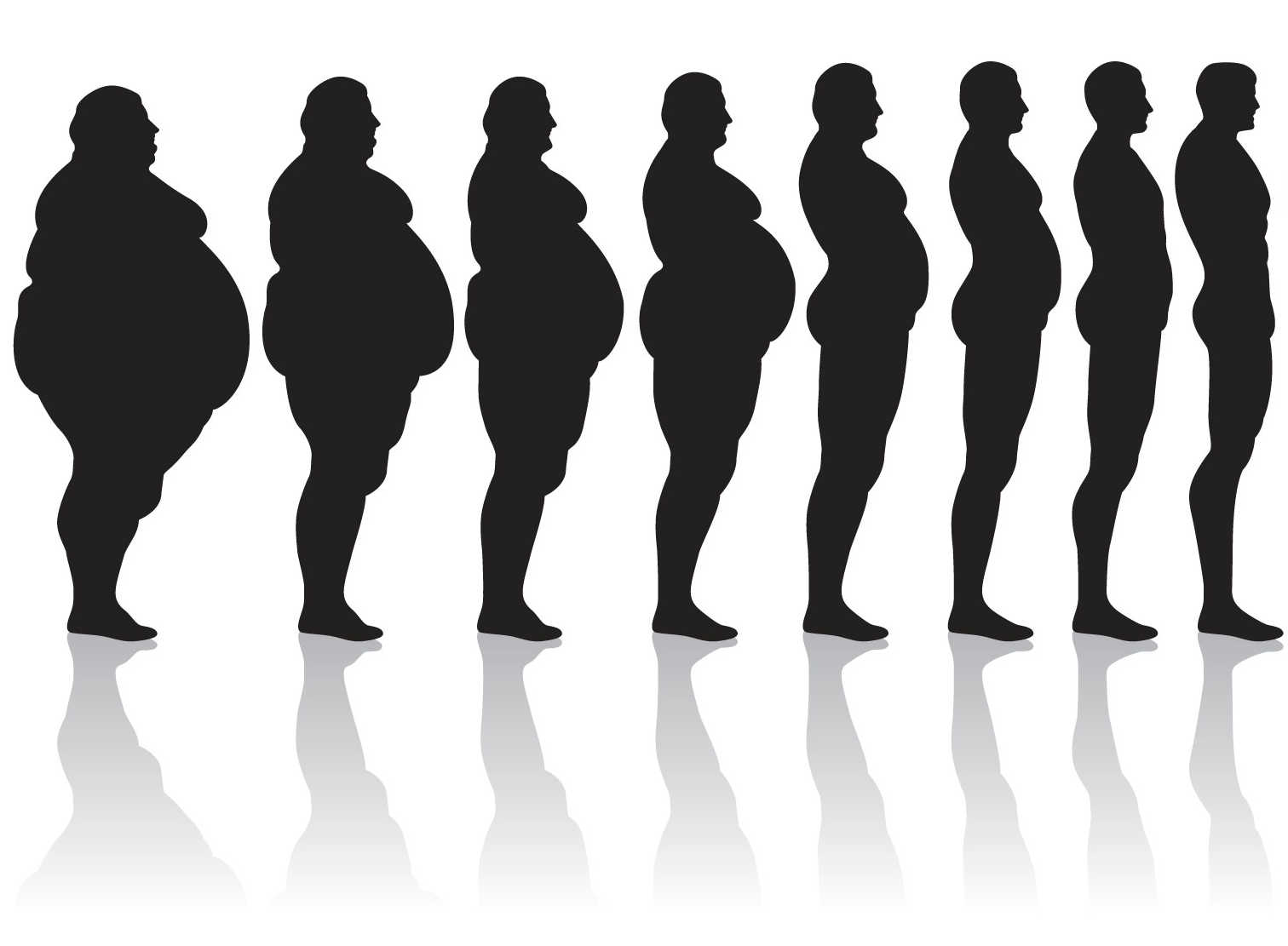


Figure 4 Weight illustration

According to this scenario two major problems can be identified that the final software has to compete with.

**Problem 1:** People leave food in their refrigerator and it can become spoiled.

**Problem 2:** Healthy nutrition and managing a balanced diet.

Hence, the purpose of this project is to develop a mobile application which supports different mobile operating systems and is able to analyze the data generated by a smart refrigerator. By means of this data base, it is capable of reminding the user in case of forgotten food which is going to spoil. Besides, it has to generate a balanced and healthy shopping list that saves time and money.

# Business Case

## Profits

As Smartphones and the Internet of things are further developing, the effect it has on our daily life is growing, too. Not only is this trend leading to a more “intelligent” life, but it also means that companies who are capable of seizing this opportunity can enhance their business.

For refrigerator manufacturers, this implies that there is a revolution pending and it is affecting the traditional refrigerator as well. Smart refrigerators are only a further development if manufacturers want to keep up with high sales and profit.

Customers can only profit from this development, not only do they invest in a new device by buying a smart refrigerator, but also into a healthy diet, prevention of food spoiling and better household management overall.

## Safety

The software comes with a familiar device (refrigerator and Smartphone), so it is not packaged as a strange and new thing that the costumer has to adjust to. Although the Internet of things is a rather new development, there are already some matured products and components. Smartphones and their applications are already an integral part of our daily lives. We use applications to manage our bank account, to listen to music, to watch videos, to communicate with friends, and even to do business.

In terms of customer and market acceptance the risks are therefore rather small. Smart refrigerator are basically regular refrigerators that have been enhanced with sensors and network connectivity, so R&D investments are rather small. Simplified, Smart refrigerators have the same benefits as regular ones but pose a supply to future demand.

## Benefit to the community

The Smart shopping list feature bears a number of benefits to its users. Users do not have to create a shopping list anymore, before heading to the supermarket. Due to heuristic shopping lists, users-behavior is affected in a subliminal way and maintenance of the inventory list according to mentioned criteria guaranteed. Over time, patterns of food consumption will become more evident in the data and thus household management more efficient.

# Risk Management

Every aspect of software project management involves some form of risk management. These possible risks have been identified and the following mitigation plans have been considered.

**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**: Development for the two main mobile platforms are quite different. Developing for two different platforms simultaneously is to expensive

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

**Risk 3:** Lack of Quality Assurance team, the software quality may not meet the requirements.

**Plan to mitigate:** 1. several test phases (Development, Alpha, Beta etc.) 2. Outsource the test.

**Risk 4:** High amount of requirements and coping with requirements changes

**Risk 5**: Short time budget; adaption to high number of different android phones.

In order to achieve effective risk management a combination of strategic (that is, planning to avoid risk) and tactical (that is, preparing to take remedial action if adverse events occur) methods was preferred. Two risk mitigation methods were considered.

## 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, the focus is on the top three risks. Each mitigation plan will be considered individually in order to mitigate the corresponding risk. Other and new occurring risks are monitored throughout the project.

## 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 of the lack of a current history data, Chaos report data was used. Budget for the risk management activity has been set aside.

According formula 2 and data from chaos report, B = 1.7Ee, Rj=0.39. An additional 0.39B are considered for Risk mitigation

Finally it must be considered, that no matter how much the chances of mitigating the impact of adverse events are increased, it cannot be guaranteed that this avoids them altogether. So, constant monitoring of risks during the whole lifecycle of the development is indispensable.

# Cost Estimate

## Labor cost

The project costs have been calculated according to the following roles and costs:



Labor costs for the whole project have been estimated (based stated salaries) as follows on the next page:

This leads to a **total labor cost** of **€ 53.776,00.**



## Overhead Expenses

Overhead expenses are listed and added to direct labor costs for four months of development. The **Overhead Expenses** are **estimated** at **€ 6225,08**. The **detailed calculations** are shown **below:**



## Project Cost

A **general and administrative expense** of **25%** **of labor and overhead costs** is **assumed**, so according to this **estimation** the **total** amount of **project cost** stands at around **€ 75.001,35**.



## Microsoft Project

Considering previous figures the following cost plan was obtained with Microsoft Project.

Overhead and general and administrative expenses were added at the end of the project during the deployment phase.

**Distribution of the labor cost regarding different labor resources:**

 **Estimated project costs divided into tasks and costs per phase:**

As common in software project high labor costs occur during testing.

**Comparison of Costs per person-month (Cocomo II) against working costs (MS-Project)**

## Function Point Analysis with COCOMO II

For the estimation of the project effort we used another method called Function Point Analysis together with the COCOMO II tool (<http://csse.usc.edu/tools/COCOMOII.php>). To calculate the parameters for this approach we used the COCOMO II Model Definition Manual (<http://csse.usc.edu/csse/research/COCOMOII/cocomo2000.0/CII_modelman2000.0.pdf>). Furthermore we calculated the Function Points with a webtool from the University of Michigan (<http://groups.engin.umd.umich.edu/CIS/course.des/cis525/js/f00/artan/functionpoints.htm>). Before that we needed basic requirements for our project to use this guidline and tools.

**Requirements for Function Points Analysis**

• User can register an account to be linked to the refrigerador.

• User can add shopping lists.

• User can remove shopping lists.

• User can modify shopping lists-

• User can retrieve the content of the refrigerator.

• User can get automatically generated shopping lists, based on the content of the refrigerador or a diet.

**Calculation of Function Points**

We used the webtool from the University of Michigan with the following parameters.





F1. Does the system require reliable backup and recovery? 3

F2. Are data communications required? 5

F3. Are there distributed processing functions? 5

F4. Is performance critical? 5

F5. Will the system run in a existing, heavily utilized operational environment? 0

F6. Does the system require on-line data entry? 3

F7. Does the on-line data entry require the input transaction to be built over

multiple screens or operations? 5

F8. Are the master files updated on-line? 5

F9. Are the inputs, outputs, files or inquiries complex? 2

F10. Is the internal processing complex? 5

F11. Is the code designed to be reusable? 5

F12. Are conversion and installation included in the design? 5

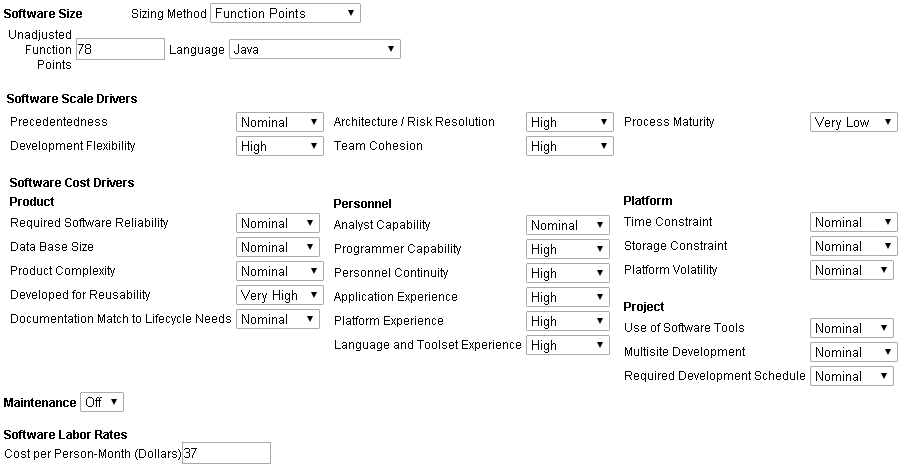
F13. Is the system designed for multiple installations in different organizations? 5

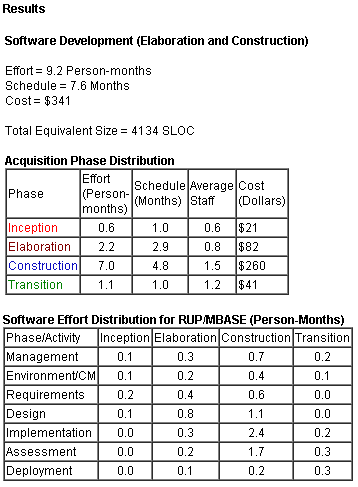
F14. Is the application designed to facilitate change and ease of use by the user? 5

According to the input our project has 78 Function Points.

**COCOMO II**

The COCOMO II tool was then fed with the 78 Function Points and more parameters according to our project and the COCOMO II manual, i.e. the average cost per person per month.





And the tool calculated following results:

For us the most interesting result is the effort in person-months to compare it to our estimation approach with Microsoft Project. The cost value can differ a lot depending on the country and business.

According to Microsoft Project our effort for the project is about 1500 hours of work taking all our resources into account.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Start** | **Finish** | **Effort** |
| Stefan Vikoler | Mon 04.01.16 | Thu 21.04.16 | 124,8 hrs |
| Filip Michnik | Thu 07.01.16 | Fri 22.04.16 | 384 hrs |
| Wang Yongjian | Thu 28.01.16 | Thu 21.04.16 | 344 hrs |
| Udeshkumar Ganesan | Fri 12.02.16 | Wed 20.04.16 | 312 hrs |
| Amélie Lagadec | Wed 20.01.16 | Wed 20.04.16 | 336 hrs |

1500 hours of work are approximately 8,5 person-months. COCOMO II calculated 9,2 person-months, which is very close to our other estimation. Besides we do not know, how and if COCOMO II calculates overhead expenses. In this case the two results are even closer.

## Earned Value Management

The Earned Value Management follows the following principles:

* Relate time phased budgets to specific contract tasks and/or statements of work.
* Provide the basis to capture work progress assessments against the baseline plan.
* Relate technical, schedule, and cost performance.
* Provide valid, timely, and auditable data/information for proactive project management analysis and action.
* Supply managers with a practical level of summarization for effective decision making.

Earned Value Management approach in this project:

1. Summarize Direct Costs by OBS Elements
2. Record/Allocate Indirect Costs
3. Identify Unit and Lot Costs
4. Track and Report Material Costs and Quantities
5. Calculate Schedule Variance and Cost Variance

The following information will be collected on a monthly basis:

1) Comparison of the amount of planned budget and the amount of budget earned for work accomplished. This comparison provides the schedule variance.

2) Comparison of the amount of the budget earned and the actual (applied where appropriate) direct costs for the same work. This comparison provides the cost variance.

1. Identify Significant Variances for Analysis
2. Analyze Indirect Cost Variances
3. Implement Corrective Actions
4. Revise Estimate at Completion (EAC)

# 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 Laws

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.

According to the EU’s Data Protection Directive all “entities or persons which collect or process personal data” [11] are “data controllers” and thus have to follow the following four principles:

* 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.” [11]

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. According to an IBM Study on data breach the average cost of each lost or stolen record is about $154 in 2015. [12]

## 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

The software company offers a basic infrastructure and developing techniques for this project.

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 necessity, the team will be expanded.

## Briggs Meyers compatibility matrix

The Myers–Briggs Type Indicator (MBTI) is an introspective self-report questionnaire designed to indicate psychological preferences in how people perceive the world and make decisions.

The 16 types are typically referred to by an abbreviation of four letters : extraversion (E), introversion (I) / intuition (N), sensing (S) / thinking (T), feeling (F) / judgment (J), perception (P).

Team member 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  -Project management | -IT Department  -Project manager | -Service Desk  employees  -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 management | -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. [13] [14]

## Functions to be tested

1. Graphical User Interface

2. Reports Output/Data

3. Report Setup/Locations

## Functions not to be tested

None

## 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).

Proper application navigation, data entry, processing, and retrieval is ensured 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.

Navigation is verified through the application and checking if it 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, an interface implemented to read the data given from the object is assumed. 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 this objective all requirements have to be specified. This **planning phase** is estimated by **19 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 the **design and prototype** of the application (with regards to the specified requirements) is scheduled. The calculated time period is **40 days.**

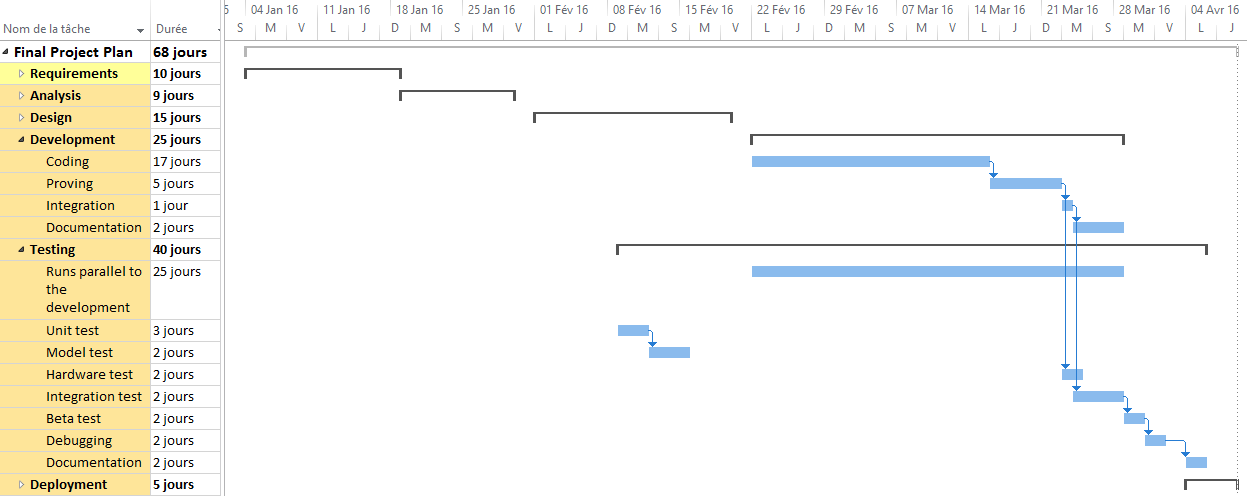
With termination of the first full functional app, improvement of all **designs and functionality** starts, including perform unit testing and acceptance testing until obtaining a final prototype and first productive application that can be launched. This phase is estimated to last **40 days.**

Then the project evolves to the support and maintenance phase. Extension of the application is possible.

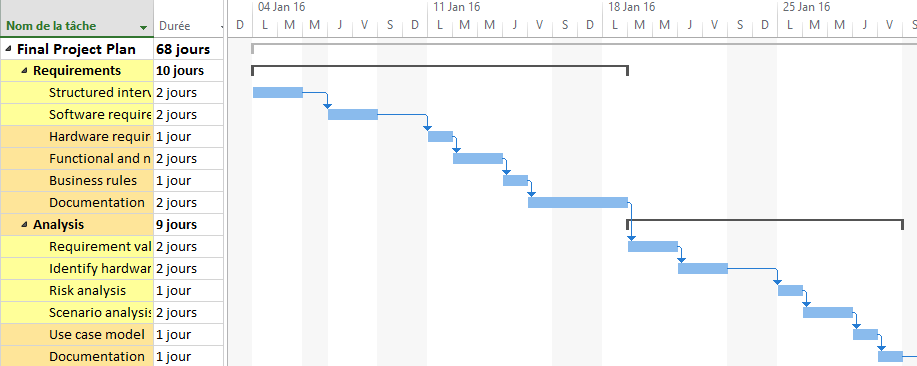
In **total**, the development of an application suited for a smart home object is assessed to be **68 days**.

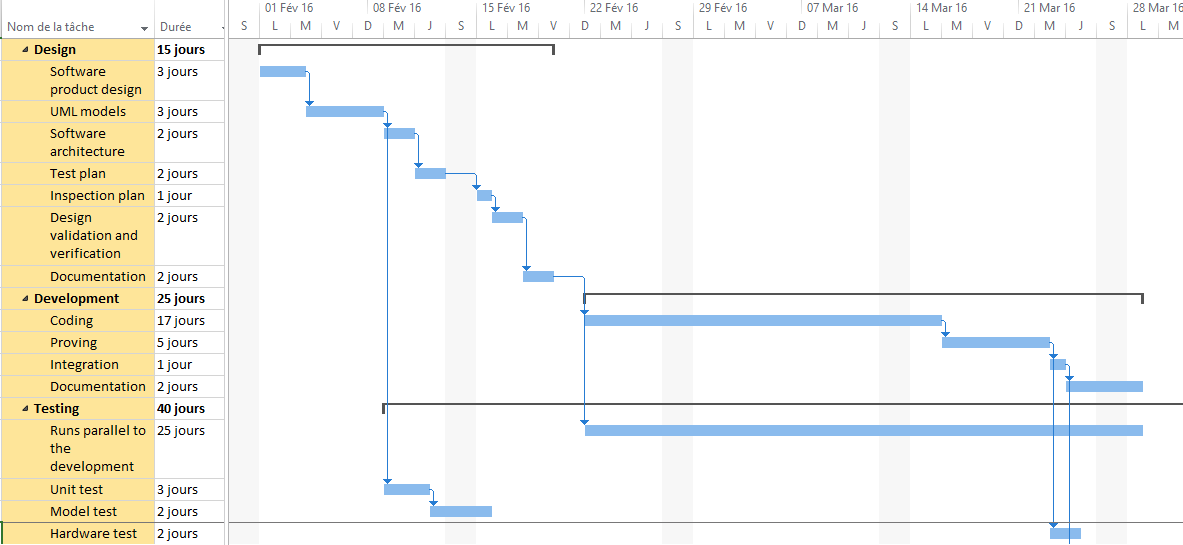
The **project** will **end** up on the **5th of April 2016.**

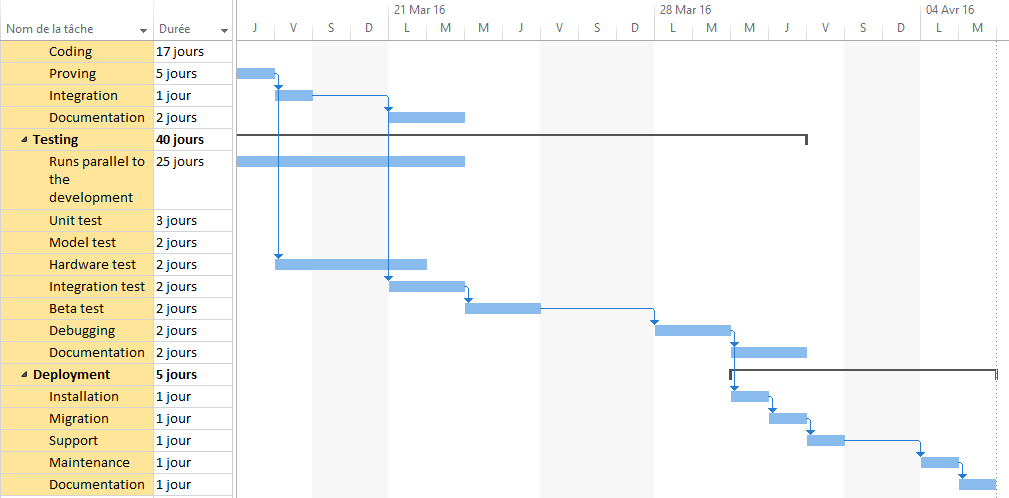
**Gantt chart overview:**



**Gantt chart (detailed)**







# Design structure matrix

The Design Structure Matrix (DSM) has the aim to perform both the analysis and the management of complex systems. It enables the user to model, visualize, and analyze the dependencies among the entities of any system and derive suggestions for the improvement or synthesis of a system.

Design Structure Matrix for this project:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Requirements | Analysis | Design | Development | Testing | Deployment |
| Requirements |  |  |  |  |  |  |
| Analysis |  |  |  |  |  |  |
| Design |  |  |  |  |  |  |
| Development |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |
| Deployment |  |  |  |  |  |  |

# References

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

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

[3] see as well 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]https://www.humphreys-assoc.com/evms/basic-concepts-earned-value-management-evm-ta-a-74.html

[11] http://ec.europa.eu/justice/data-protection/data-collection/index\_en.htm

[12] http://www-03.ibm.com/security/data-breach/

[13] 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

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

1. according to a recent PwC Study [5] [↑](#footnote-ref-1)
2. 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. [↑](#footnote-ref-2)