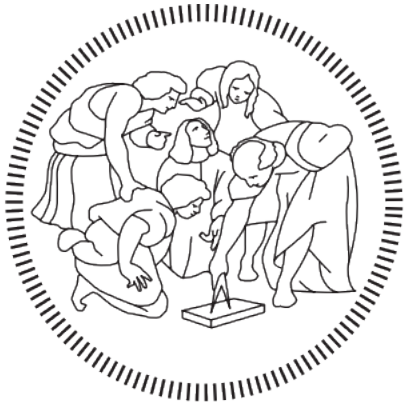
PowerEnJoy project: Project Plan Document

Boriero Stefano 876106 Brunitti Simone 875039

January 18, 2017



POLITECNICO
MILANO 1863

Contents

1 Introduction

2 Project Size and Cost Estimation

2.1 Size estimation: function points

2.1.1 Internal Logic Files

In order to provide its functionalities, PowerEnjoy will need to store data and information.
The main information it need to store is about vehicles. The relevant attributes needed are

- Vehicle Id
- Plate number
- Status
- BatteryLevel
- Position

Another fundamental table is the one about rides, containing this information:

- Starting point
- Starting battery
- Number of passengers
- Final position
- Final battery

To complete the set of tables oriented to the actual sharing functionality, the system will have a Reservation table with the following information:

- User id
- Vehicle id
- Starting time
- Unlocked car

For assistance list related functionalities the system needs an assistance table to store:

- Vehicle id
- Fault

- Status
- Employee in charge

Employee's information are stored as well:

- Employee id
- Name
- Surname
- District
- Job

Then the system needs all usual information about users:

- User id
- Username
- Name
- Surname
- Driving license

For security issues, payment information are stored in a different table:

- User id
- Payment info

Safe areas are store in a two-level structure. The first one containig:

- SafeArea id
- Type

The second one containing:

- SafeArea id
- Position

The following table sums up ILF's complexity and FP associated

ILF	**Complexity**	**FP**
Login data	Low	7
User data	Low	7
Employee data	Low	7
Password	Average	10
Safe areas	Low	7
Vehicle	Low	7
Reservation	Low	7
Ride	Average	10
Assistance	Low	7
Total		69

2.1.2 External Logic Files(ELFs)

The first external data source of PowerEnJoy is Google Maps API. The interactions between our system and this API are the following:

- **Map Displaying**: The API must interact with the customer's terminal device in order to display a map showing the nearest cars and their distance in an area selected by the user. This first interaction is made quite simple by Google Maps API, so we will give it an average complexity
- **Drawing Tools**: The API must interact with the employee system to allow the employee to draw new safe areas and see the existing ones. This interaction is quite complex, so we will give it an high complexity.

The following table sums up ELF complexity and FP associated

ELF	**Complexity**	**FP**
Map Displaying	Average	7
Drawing Tools	High	10
Total		17

2.1.3 External Inputs (EIs)

In this section we will list all the different kind of interactions that PowerEnJoy has with its users. First, we list the interactions that any user can have with the platform:

- Login/Logout: simple operations, involve only the Account Controller. Points = 2x3FPs.

Then, we list the interactions between the customer and the platform:

- Register a new account: Since this feature requires some data to be checked, we give it an Average complexity. Points = 4FPs.
- Rent a car : This is a key feature and a highly complex one, involving many different components. Points = 6 FPs.
- Add money to the balance: For security reasons, this operation requires some steps. We give it average complexity. Points = 4FPs.
- Edit personal info: Some validation of the new data is required, therefore it's average complexity. Points = 4FPs
- Report a damage: this is a simple task, involving only one component. Points = 3FPs.

Finally, we list the interaction that an employee can have with the platform:

- Drawing a new safe area: Since the process requires the area to be valid (and the validity of an area needs to be checked), we think this is a complex opearation. Points = 6FPs.
- Taking in charge a vehicle: This is a straightforward task and a fairly simple one. Points = 3FPs.

- Register/Dismiss a car to the fleet: Since this operation requires cooperation between the car and the central system, we conside it to be complex. Points = 6x2 FPs

The following table sums up EIs complexity and FP associated:

EI	**Complexity**	**FP**
Login	Low	3
Logout	Low	3
Register new account	Average	4
Rent a car	High	6
Add money	Average	4
Edit personal info	Average	4
Report a damage	Low	3
Drawing new safe area	High	6
Take in charge vehicle	Low	3
Register a car	High	6
Dismiss a car	High	6
Total		48

2.2 External Inquiries

The definition of external inquiry is a data request performed by a user. We divide have two different type of users, customers and employees. For the first we have the following possible inquiries:

- Get personal information
- Get available vehicles from a given position

For the second we have:

- Get Safe areas
- Get assistance list

Getting the available vehicles requires a bit of computation regarding vehicles position, so it will be labeled with an Average complexity. All the other interactions can be accomplished by a single query each, so they're classified with a Low complexity. A brief recap shown in the following table:

External Inquiry	**Complexity**	**FP**
Get personal information	Low	3
Get available vehicles	Average	4
Get safe areas	Low	3
Get assistance list	Low	3
Total		13

2.3 External Outputs

The system need also to communicate with users other information than results of inquiries. These operations go under the name of External Outputs, and occurr on these occations:

- Notify a user he's a Debtor
- Notify a user he's been Banned
- Notify a user he committed an Infraction
- Notify a user his payment outcome
- Notify an employee a car has been added to assistance list

Since the Notification Gateway our system is going to use is an external one, all these operations have a low complexity in our development process.

External Outputs	**Complexity**	**FP**
Notify user he's a Debtor	Low	4
Notify user he's been Banned	Low	4
Notify user he committed an Infraction	Low	4
Notify user his payment outcome	Low	4
Notify employee a car has been added to assistance list	Low	4
Total		20

2.4 Overall Estimation

A recap of everything we have stated about the complexity of our system is provided by this table

External Outputs	**FP**
Internal Logic File	69
External Logic File	17
External Inputs	48
External Inquiries	13
External Outputs	20
Total	167

Considering Java Enterprise Edition as a development platform and disregarding the aspects concerning the implementation of the mobile applications (which can be thought as pure presentation with no business logic), we can estimate the total number of lines of code. Depending on the conversion rate, we have a lower bound of: $SLOC = 167 * 46 = 7682$ and an upper bound of: $SLOC = 167 * 67 = 11189$

2.5 Cost and effort estimation: COCOMO II

2.5.1 Scale Drivers

Scale Factors	Very Low	Low	Nominal	High	Very High	Extra High
PREC	thoroughly unprecedented	largely unprecedented	somewhat unprecedented	generally familiar	largely familiar	thoroughly familiar
SF_j	6.20	4.96	3.72	2.48	1.24	0.00
FLEX	rigorous	occasional relaxation	some relaxation	general conformity	some conformity	general goals
SF_j	5.07	4.05	3.04	2.03	1.01	0.00
RESL	little (20%)	some (40%)	often (60%)	generally (75%)	mostly (90%)	full (100%)
SF_j	7.07	5.65	4.24	2.83	1.41	0.00
TEAM	very difficult interactions	some difficult interactions	basically cooperative interactions	largely cooperative	highly cooperative	seamless interactions
SF_j	5.48	4.38	3.29	2.19	1.10	0.00
PMAT	Level 1 Lower	Level 1 Upper	Level 2	Level 3	Level 4	Level 5
SF_j	7.80	6.24	4.68	3.12	1.56	0.00

- **Precedentness**: represents the level of experience our team has in developing large projects. Since this is the first project of such dimension we're facing, this value will be low.
- **Development Flexibility**: represents the degree of flexibility in the development process with respect to external specification and requirements. Since the stakeholders do not exist, we assume requirements as immutable, thus the value will be low.
- **Risk resolution**: represents how much we are aware of risks and how we are ready to react if anything happens. Since we have performed a good risk analisys but not much extensive, we will set this value to Nominal
- **Team Cohesion**: represents the ability of team members to work togheter and know each other. Since we had exceprience in the development of another project last year, this value is set to very high.
- **Project Maturity**: represents the level of progress in the system project. Since we have already produced RASD, DD and ITPD, this is set to level 4.

Here's a tabular recap of what stated above

Scale Driver	Factor	Value
Precedentness(PREC)	Low	4.96
Development Flexibility(FLEX)	Low	4.05
Risk Resolution(RESL)	Nominal	4.24
Team Cohesion(TEAM)	Very High	1.10
Process Maturity(PMAT)	Level 4	1.56
Total		15.91

2.5.2 Cost Drivers

- **Required Software Reliability**:
  Since there are many car sharing services, a malfunctioning would lead to a loss of customers for the company, along with a bad reputation. This can be translated into a high financial lost, so the Rating Level is set to High

RELY Cost Drivers						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	Slightly inconvenience	Easily recoverable losses	Moderate recoverable losses	Highly financial loss	risk to human life	
Effort multiplier	0.82	0.92	1.00	1.10	1.26	n/a

- **Database size**:
  Given the aforementioned tables and attributes, we estimate roughly the dimension of database to be around 5GB: since we estimate KSLOC to be between 8 and 11, the ratio D/P is between 450 and 625. The Rating Level is set to High.

DATA Cost Drivers						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor		$D/P < 10$	$10 \leq D/P \leq 100$	$100 \leq D/P \leq 1000$	$D/P > 100$	
Effort multiplier	n/a	0.90	1.00	1.14	1.28	n/a

- **Product Complexity**:
  According to COCOMO II CPLX table, this Rating Level is set to Nominal.

CPLX Cost Drivers						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Effort multiplier	0.73	0.87	1.00	1.17	1.34	1.74

- **Required Reusability**:
  the company might want to extend the car sharing service to other types of vehicle, so some components should be developed as reusable but only in the project scope. The Rating Level is thus set to Nominal

RUSE Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor		None	Across project	Across program	Across product line	Across multiple product lines
Effort multiplier	n/a	0.95	1.00	1.07	1.15	1.26

- **Documentation match to life-cycles**:
  Every document needed for product life-cycles has been developed, so we set this Rating Level to Nominal

DOCU Cost Drivers						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	Many life-cycle needs uncovered	Some life-cycle needs uncovered	Right sized to life-cycle needs uncovered	Excessive for life-cycle needs uncovered	Very excessive for life-cycle needs uncovered	
Effort multiplier	0.81	0.91	1.00	1.11	1.23	n/a

- **Execution time constraint**:
  As PowerEnJoy will envolve a fairly usage of the CPU available time, due to both the complexity of the system and the high level of traffic generated by both users and cars, the Rating Level is set to Very High

TIME Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor			$\leq$ 50% use of available execution time	70% use of available execution time	85% use of available execution time	90% use of available execution time
Effort multiplier	n/a	n/a	1.00	1.11	1.29	1.63

- **Storage Constraint**:
  Current disk drives offer a more than suitable capacity for the need of our application. However, an increase in the number of vehicles could require more and more data to be stored, so we overestimate this Rating Level to High

STOR Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor			$\leq$ 50% use of available storage	70% use of available storage	85% use of available storage	90% use of available storage
Effort multiplier	n/a	n/a	1.00	1.05	1.17	1.46

- **Platform Volatility**:
  We don't expect major changes to happen often, as the additional functionalities for such an application are limited; plus, as we don't plan on

providing a dedicated app, even client side won't need much changes to remain up to date. For these reasons, the Rating Level is set to Low

PVOL Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	Major changes every 12 months, minor changes every 1 month	Major changes every 6 months, minor changes every 2 weeks	Major changes every 2 months, minor changes every 1 week	Major changes every 2 weeks, minor changes every 2 days		
Effort multiplier	n/a	0.87	1.00	1.15	1.30	n/a

- **Analyst Capability**:
  we have deeply analized the problem and its real world impact, so we set this Rating Level to High.

ACAP Cost Drivers						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	
Effort multiplier	1.42	1.19	1.00	0.85	0.71	n/a

- **Programmer Capability**:
  The project must still be developed, but we have already worked together as a team, so we are confident in our capabilities. We will set this parameter to High.

PCAP Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	
Effort multiplier	1.34	1.15	1.00	0.88	0.76	n/a

- **Application Experience**:
  This is the first time we are using JEE, even if we have already worked on some java applications. We are setting this parameter to Very Low, since we have started using JEE less than 2 months ago.

APEX Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	less than 2 months	6 months	1 year	3 years	6 years	
Effort multiplier	1.22	1.10	1.00	0.88	0.81	n/a

- **Platform Experience**:
  As already said, we have little experience with JEE. We have however some

experience in databases and graphic user interfaces, so we are setting this parameter to Nominal.

PLEX Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	less than 2 months	6 months	1 year	3 years	6 years	
Effort multiplier	1.19	1.09	1.00	0.91	0.85	n/a

- **Language and Tool Experience**:
  For the reasons stated above, this parameter is set to Nominal

LTEX Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	less than 2 months	6 months	1 year	3 years	6 years	
Effort multiplier	1.20	1.09	1.00	0.91	0.84	n/a

- **Personnel Continuity**:
  Since before being programmer we are student, we can spend only part of our day in the development of this project. For this reason, this parameter is set to Very Low.

PCON Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	48 % /year	24 % /year	12 % /year	6 % /year	3 % /year	
Effort multiplier	1.29	1.12	1.00	0.90	0.81	n/a

- **Usage of Software Tools**:
  Our application platform supports basic life-cycle tools, so we will set this parameter to Nominal

TOOL Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	edit, code, debug	simple, frontend, backend CASE, little integration	basic life-cycle tools, moderately integrate	strong, mature life-cycle tools, moderately integrated	strong, mature, proactive life-cycle tools, well integrated with processes, methods, reuse	
Effort multiplier	1.17	1.09	1.00	0.90	0.78	n/a

- **Multisite Development**:
  Even if we work from two different cities, we have frequent wideband

electronic communication and occasional voice conference. Due to these factors, we are setting this parameter to Very High.

SITE Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	Some phone, mail	Individual phone, FAX	Narrowband email	Wideband electronic communication	Wideband elect. comm, occasional video conf.	Interactive multimedia
Effort multiplier	1.22	1.09	1.00	0.93	0.86	0.80

- **Required Development Schedule**:
  Since we plan to put more effort in the early part of our project, we will set this parameter to High

SCED Cost Driver						
Rating Level	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Descriptor	75 % of nominal	85 % of nominal	100 % of nominal	130 % of nominal	160 % of nominal	
Effort multiplier	1.43	1.14	1.00	1.00	1.00	n/a

We can resume cost drivers in this table

Cost Driver	**Factor**	**Value**
Required Software Reliability(RELY)	High	1.10
Database size(DATA)	High	1.14
Product Complexity(CPLX)	Nominal	1.00
Required Reusability(RUSE)	Nominal	1.00
Documentation match to life-cycles(DOCU)	Nominal	1.00
Execution time constraint(TIME)	Very High	1.29
Storage Constraint(STOR)	High	1.05
Platform Volatility(PVOL)	Low	0.87
Analyst Capability(ACAP)	High	0.85
Programmer Capability(PCAP)	High	0.88
Application Experience(APEX)	Very Low	1.22
Platform Experience(PLEX)	Nominal	1.00
Language and Tool Experience(LTEX)	Nominal	1.00
Personnel Continuity(PCON)	Very Low	1.29
Usage of Software Tools(TOOL)	Nominal	1.00
Multisite Development(SITE)	Very High	0.86
Required Development Schedule(SCED)	High	1.00
Total		17.55

3 Schedule

4 Resource Allocation

5 Risk Managment

1. **Burocracy issues**:
   One of the most dangerous risks is related to the interaction with local government. The acquiring of permissions could take a long time, depending on the city, and this could lead in a major delay in our project. To prevent this we overestimate the time needed to obtain those permissions.

2. **Data loss**:
   Data loss can be a consequence to hackers' attack or power supply issues. To avoid this problem we need to backup our data frequently on multiple external disks.

3. **Stakeholders' bankrupt**:
   Since we have been commissioned this project by a private company and all the costs are covered by them, a bankrupt of that company would mean an almost certain failure for our software company. To prevent this, we ask them a financial guarantee.

4. **Dependency on external services**:
   As we exploit external services in our project, like GoogleMaps, Payment services and Notification services, we might encounter compatibility problems if any of the external provider introduces major changes in the way services are provided. A kind of preventive solution could involve exploiting information hiding and designing components in a modular way, in order to isolate the interface with those external services and make it easier to replace.

5. **Changes in requirememnts**:
   Following a meeting with the stakeholders, some controversy concerning our interpretation of the requirements might arise. To avoid to make major changes to our project, we will schedule many meetings with the stakeholder in order to minimize incomprehensions by keeping a constant flow of information between our software house and the company.

6. **Low appeal to user**:
   Once the software is released, we will incurr in user's review of our pruduct. To address better their taste, we will release beta versions of the client side during the development process, in order to collect opinions and adjust our work prior to actual releas. This will prevent us from applyng big changes right after the actual release, and will help us to develop a better product.

6 Effort Spent