Product Prototyping and Service Definition

What does Product/Service prototyping means?

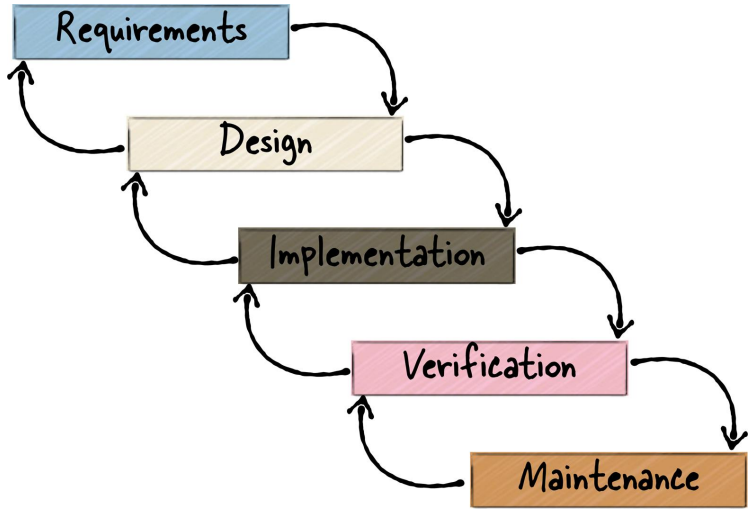
What does Product/Service prototyping means?

Rapid prototyping is a set of industrial **techniques** aimed at the «physical» **realization of a prototype**, in a relatively **short time**.

What does Product/Service prototyping means?

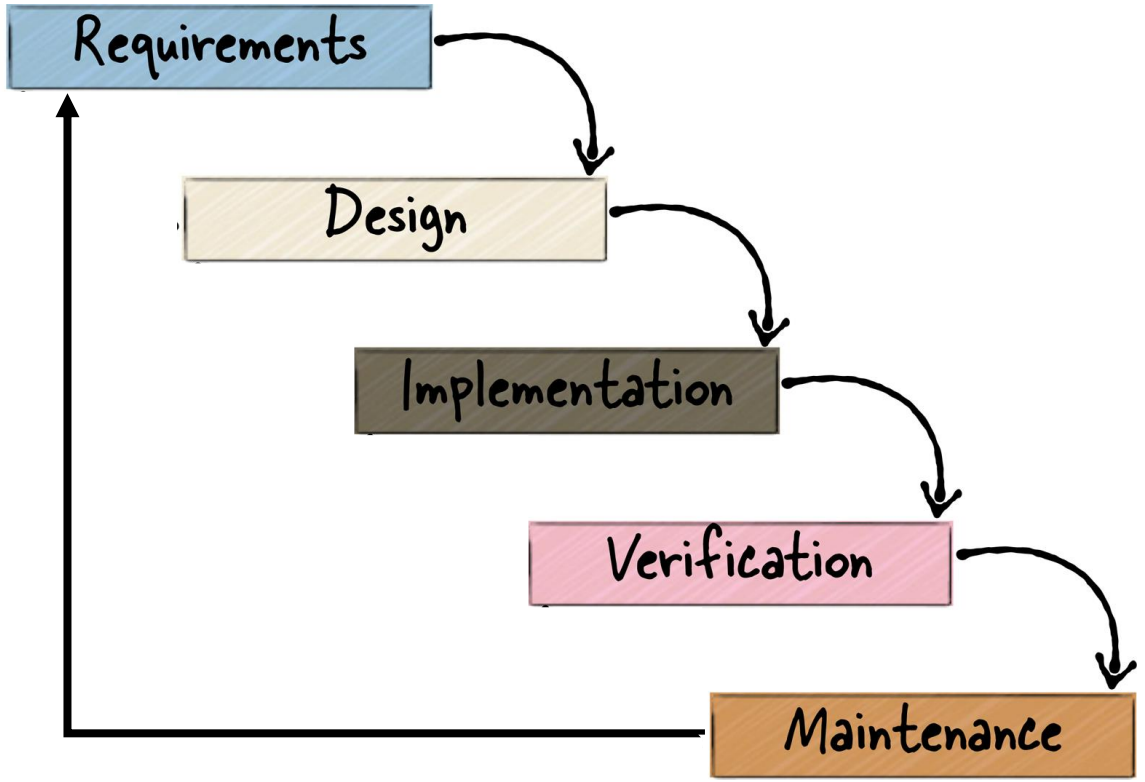
Rapid prototyping is a set of industrial **techniques** aimed at the «physical» **realization of a prototype**, in a relatively **short time**.

In **Software Engineering**, **Prototype** methodology is a **software development** model in which a **prototype** is built, test and then reworked when needed until an acceptable **prototype** is achieved.



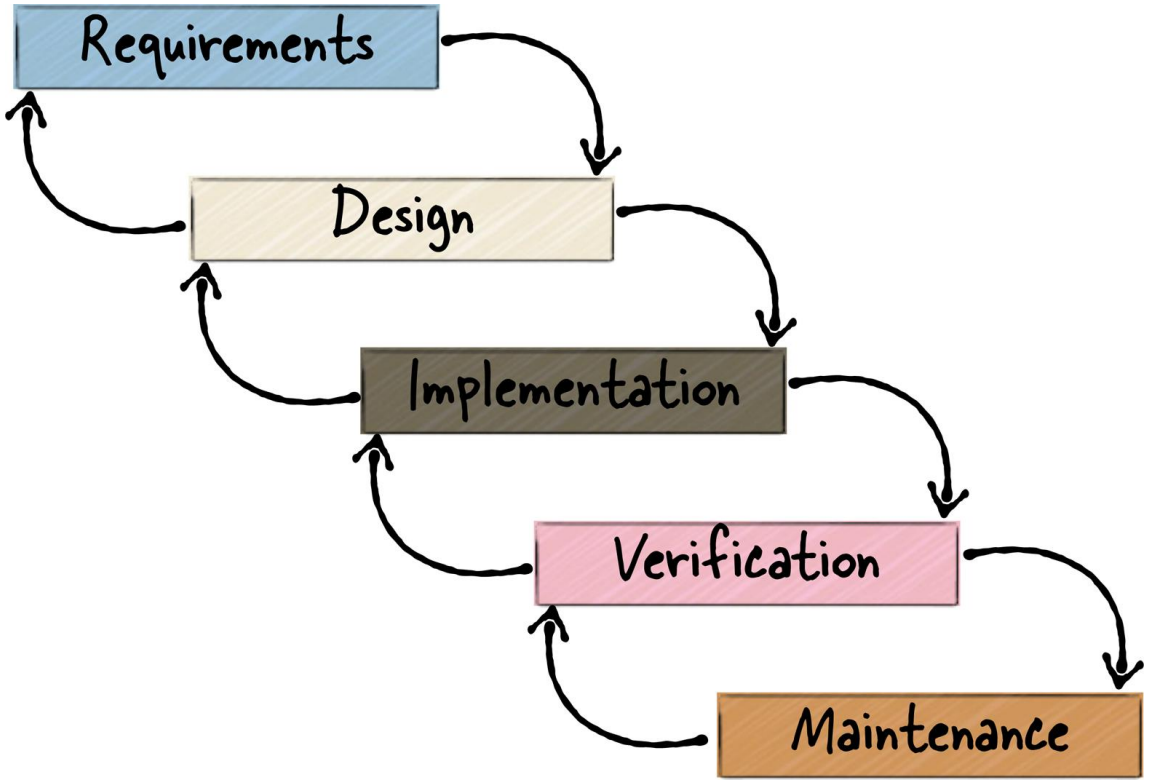
Models – Delivering a Prototype

Waterfall Model

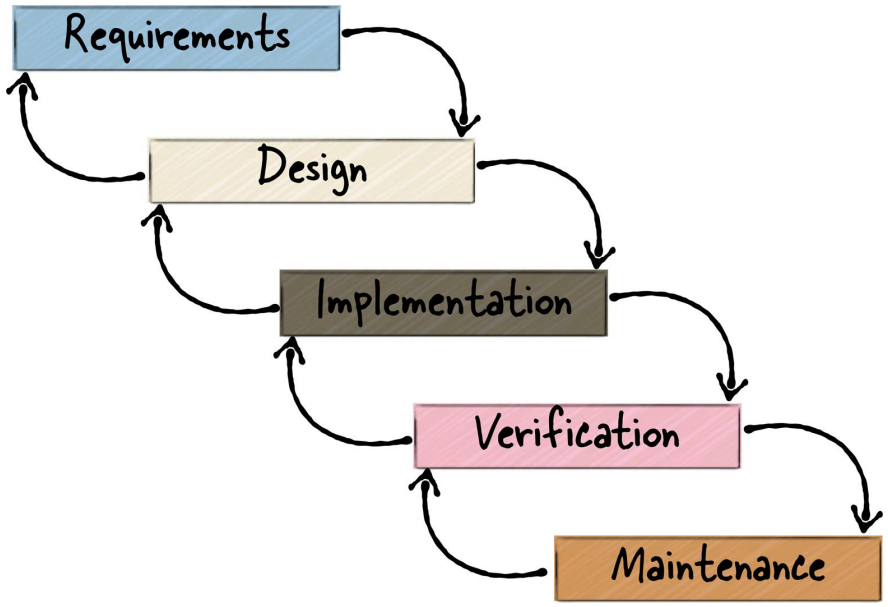


Models – Delivering a Prototype

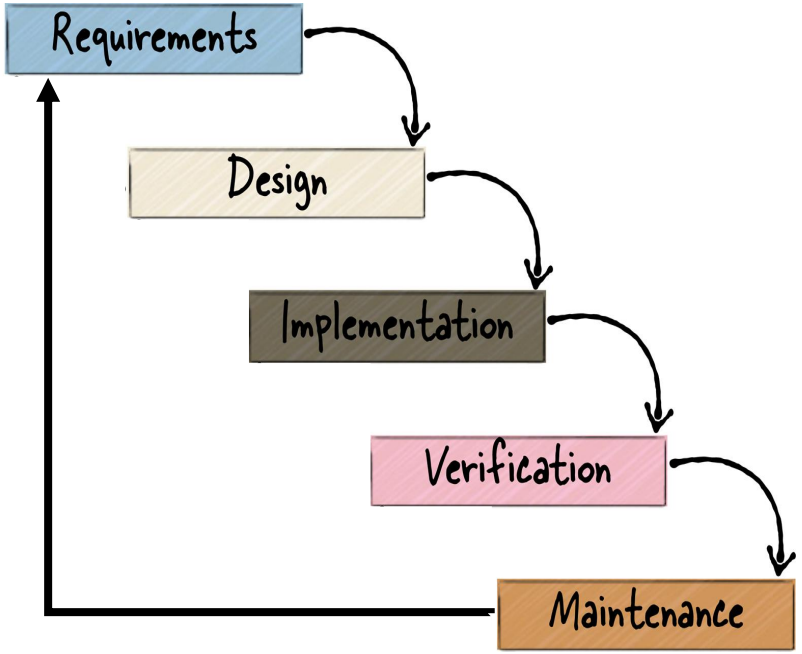
Agile



Agile VS Waterfall



VS



What is agile?

- Agile is a set of methods and methodologies
- Agile is a mindset
- Based on ideas, Values and principles

Agile values

- **Individual and interactions** over processes and tools
- **Working software/(whatever)** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following plan

Examples of Agile Methodologies-Scrum and Kanban

Giacomo’s Thesis!

Agile methodologies for an innovative start-up

Candidate:

Giacomo Plebani

SUPERVISORS:

ANDREA POLINI

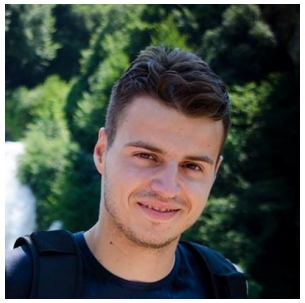
FRANCESCO PEZZUOLI

Case of study: tkp

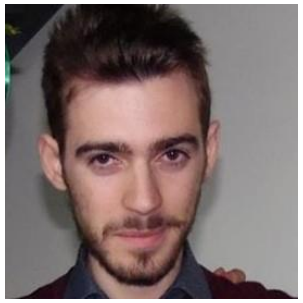
A supporting device which helps people who are unable to speak with their voice and physically disabled

Team

Francesco
Ceo Limix/product owner



Giacomo
Scrum Master



Davide
Developer

Giulia
Mathematician

Andrea
Designer

Sprint 1

The first sprint lasted 3 weeks, from 14th of February to 4th of March.

The aim was to present the first prototype to the customer and the end of the sprint.

Sprint 2

Once we had finished the prototype and get feedbacks from customer, the next step was to decide which graphic interface and keyboard design can perform better.

This sprint lasted 2 weeks, from 13th to 24th of March.

Sprint 3

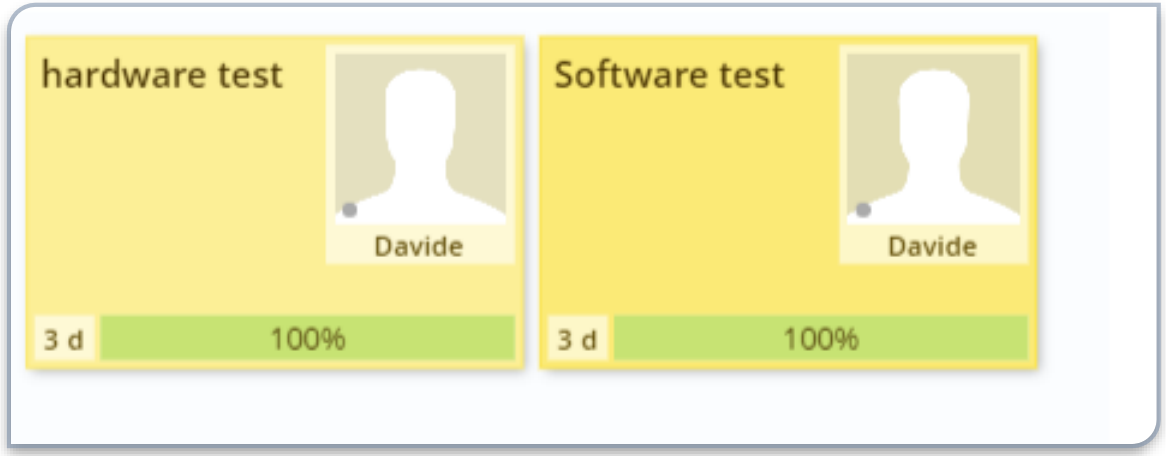
The third sprint is the core of the project, the majority of the work has been done in this sprint.

Cause of the amount of work the third sprint had a duration of 6 weeks from 27th of March to 5th of May.

Sprint 4

This is the last sprint, with focus on testing hardware and software.

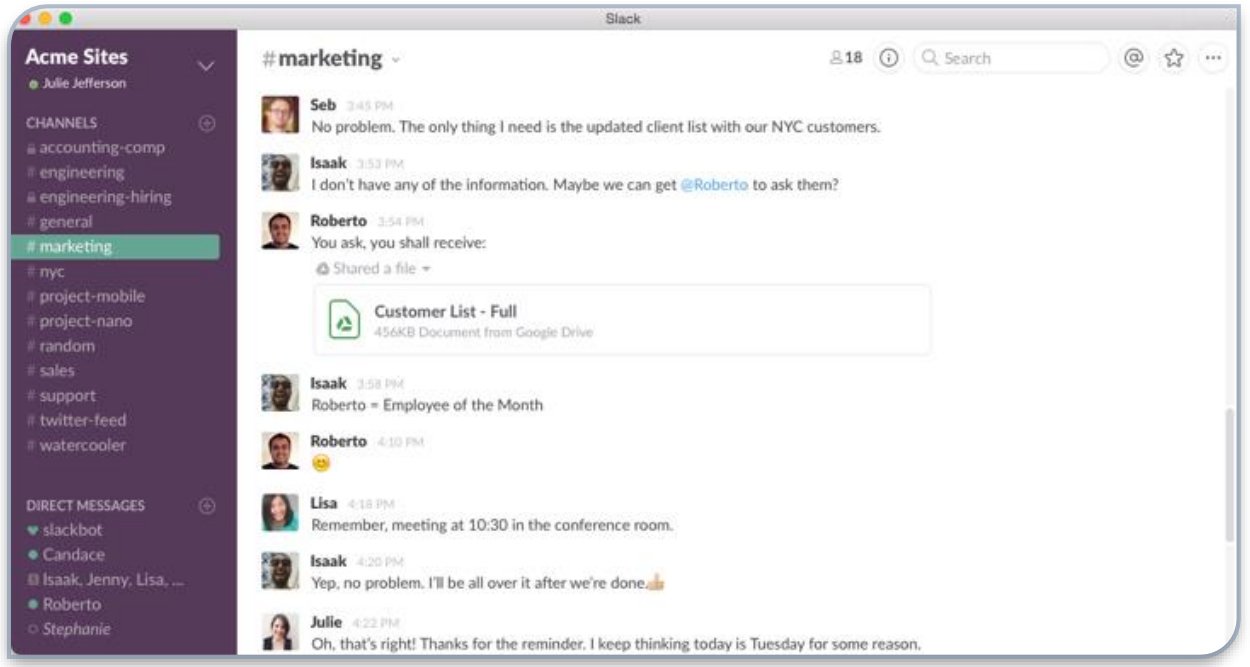
The sprint started the 29th of May and finished the 9th of June.



Slack

To manage communication and to share documents we used slack.

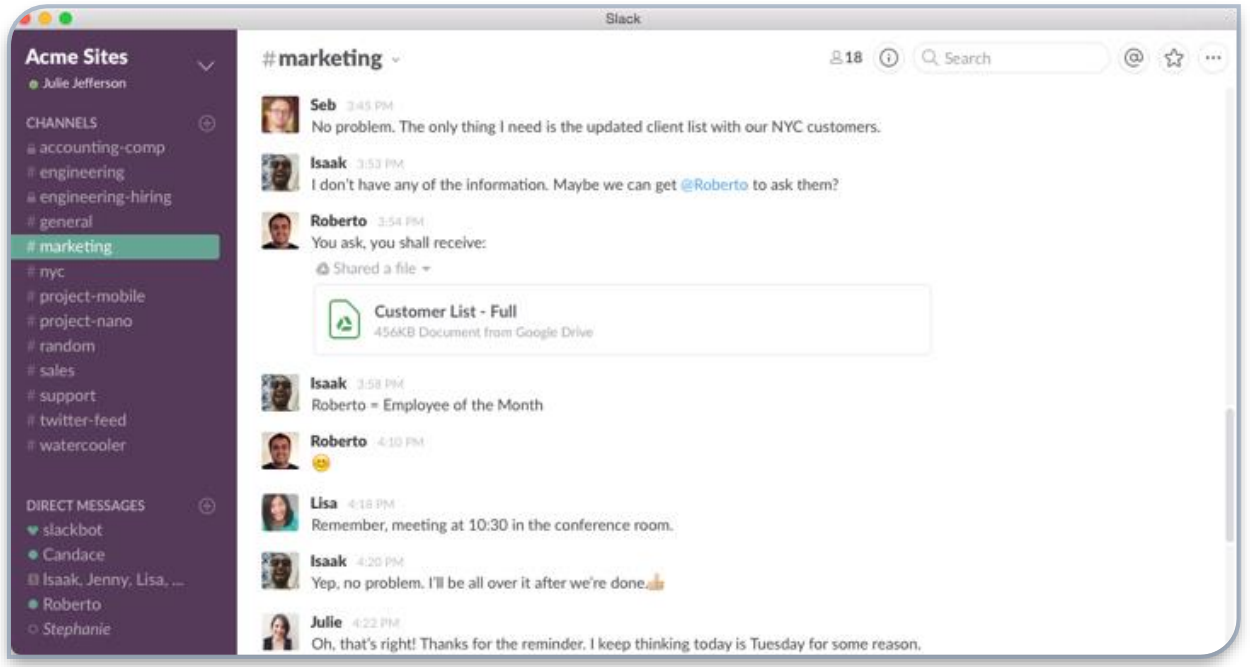
Slack is basically a messaging app on steroids. It's meant for teams and workplaces, can be used across multiple devices and platforms.



Slack

To manage communication and to share documents we used slack.

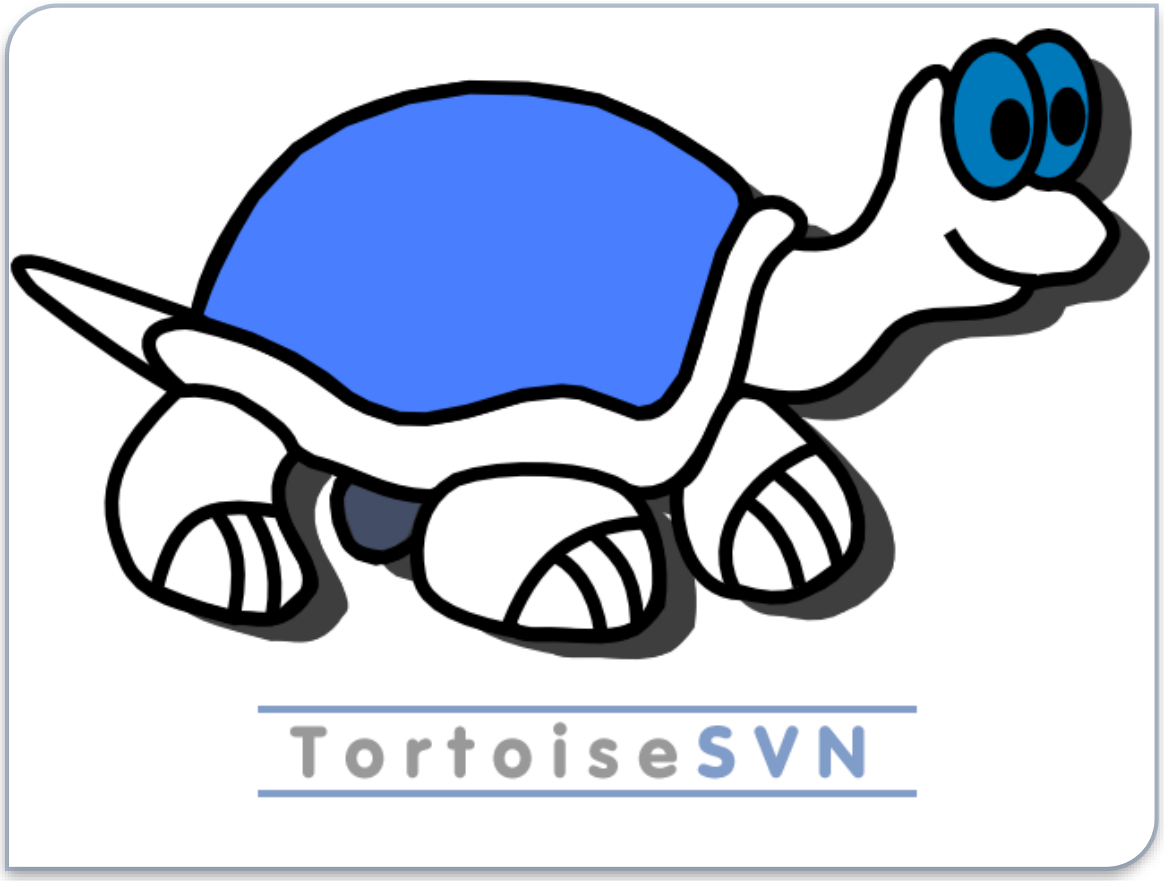
Slack is basically a messaging app on steroids. It's meant for teams and workplaces, can be used across multiple devices and platforms.



...But I really enjoy telegram

Tortoise svn

Tortoise SVN is a free open-source Windows client for the Apache Subversion version control system and manages files and directories over time. Files are stored in a central repository. The repository is much like an ordinary file server, except from the fact that it remembers every change ever made to your files and directories.



Scrumwise

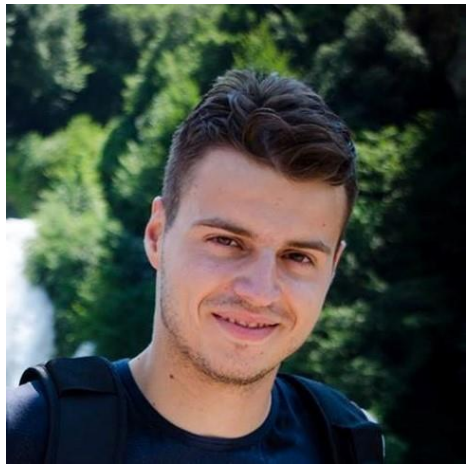
Scrumwise is an online tool which permits the management of scrum when is not possible to do in office. All the information are saved in cloud and all the members of the team can see the specification of the project.

Framework for innovative start-up

- Is it compatible with agile and scrum values?

Individuals and interactions	over	Process and tools
Working software	over	Comprehensive documentation
Customer collaboration	over	Contract negotiation
Responding to change	over	Following a plan

Giacomo’s Conclusions (in brief)



Giacomo
Scrum Master

Team building

- Team is the main resource
- Figure out how many people you need and in which fields
- Skills are important, but is better to take someone with the sense of commitment
- No matter if they don’t know agile in general
- Let them apprehend scrum little by little (analogies and examples help a lot)

The roles of scrum into a start-up

- Product owner is the responsible of the product backlog and he prioritizes which are the most valuable features. He serves as a joining link from market to development. Is better if he has engineer skills
- The scrum master of course is someone who knows scrum, and he has to ensure which scrum is adopted in a right way. In a start-up context scrum master is similar to a project manager
- Team members

Tasks vs user stories

- Tasks are better than user stories into a start-up context
  - Easier to understand
  - Smaller
  - Easy to recover the work
- Tasks are owned by the entire team, but choose a responsible for each one (in general is the member with more experience in that field)

Tasks!

- Is a requirement (or a part of it) that needs to be developed!
- A task is assigned to “someone”.
- Usually we give to a task value: “How much time needed to be completed?”
- Usually we give a priority to a task!

Requirements

- Is a **requirement** (or a part of it) that needs to be developed!
- A requirement is something needed or wanted form the customer (or the project manager)! Simple.

What I have to do with this requirements?

- Do a list of requirements!
- There are two way of talking about a requirement:
- WAY 1: Requirements elicitation/ Requirements Analysis
- WAY 2: Requirements Development

WAY 1: Requirements elicitation/ Requirements Analysis

- This is the most important!

WAY 1: Requirements elicitation/ Requirements Analysis

What the proto/system should do?

- This is the most important!

System Requirements

The table shows all the features carried out in the project:

ID	***NAME***	***PRIORITY***	***DESCRIPTION***
ID-01	***Scratches recognition***	*HIGH*	The System must detect a scratch movement using a device connected with the forearm and the hand.
ID-02	***Affidability***	*HIGH*	The System must provide reliable values with a 95% of confidence.

System Requirements

The table shows all the features carried out in the project:

ID	NAME	PRIORITY	DESCRIPTION
ID-01	**Scratches recognition**	*HIGH*	The System must detect a scratch movement using a device connected with the forearm and the hand.
ID-02	**Affidability**	*HIGH*	The System must provide reliable values with a 95% of confidence.
ID-03	**False Positive Strategy**	*HIGH*	If the User is scratching, the System gives a feedback and then the User can eventually delete the last detected data.
ID-04	**Battery supply**	*MEDIUM*	The System can be battery powered.
ID-05	**Scratches Feedback**	*MEDIUM*	The System must give a feedback when the User scratches.
ID-06	**Low Battery**	*MEDIUM*	The System must make a feedback when it is close to power off.
ID-07	**Data User Management**	*MEDIUM*	If the System is powered by the PC, the User can save data in a file.
ID-08	**Ergonomics**	*LOW*	The System must be easy to put and take off.

University of Camerino

SCHOOL OF SCIENCE AND TECHNOLOGY

Master of Science in Computer Science (LM-18)

EMBEDDED SYSTEM PROJECT

Itchy and Scratchy

Scratch recognizer using BNO055 and MyoWare Muscle Sensor

Students

Margherita Renieri

Corrado Petrelli

Luca Marasca

Professor

Prof. Francesco Pezzuoli

ACADEMIC YEAR 2018/2019

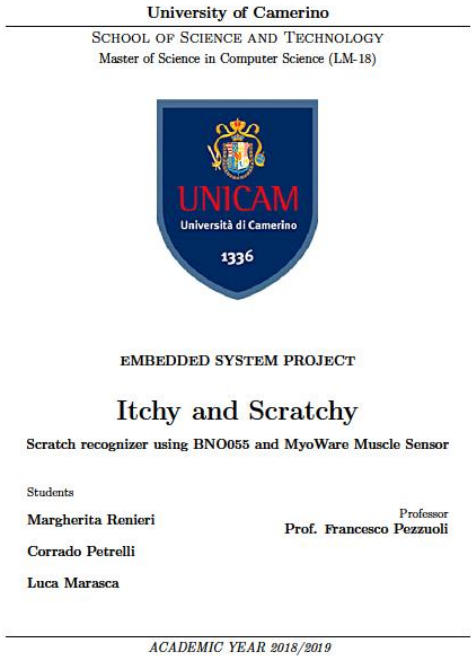
DURING Requirements elicitation/ Requirements Analysis

YOU DON'T HAVE TO TALK ABOUT HOW YOU WANT TO DEVELOP A REQUIREMENT!!!

WAY 2: Requirements Development

How I deploy my proto/system?

- This is also important! ☺



University of Camerino

SCHOOL OF SCIENCE AND TECHNOLOGY

Master of Science in Computer Science (LM-18)

EMBEDDED SYSTEM PROJECT

Itchy and Scratchy

Scratch recognizer using BNO055 and MyoWare Muscle Sensor

Students

Margherita Renieri

Corrado Petrelli

Luca Marasca

Professor

Prof. Francesco Pezzuoli

ACADEMIC YEAR 2018/2019

Let’s Write some Requirement of your project

System Requirements

The table shows all the features carried out in the project:

ID	***NAME***	***PRIORITY***	***DESCRIPTION***
ID-01	***Scratches recognition***	*HIGH*	The System must detect a scratch movement using a device connected with the forearm and the hand.
ID-02	***Affidability***	*HIGH*	The System must provide reliable values with a 95% of confidence.

Practical

- Requirements Development:
  - Technology – Embedded Systems
  - Design – Basics for an usable product

Embedded Systems

What is an Embedded System?

What is an Embedded System?

- Integrated system,
- Usually contains a microprocessor,
- Manage Signals
- And…?

What is an Embedded System?

- Integrated system,
- Usually contains a microprocessor,
- Manage Signals
- Specific Purpose! (usually)

What is an Embedded System?
-Examples

What is an Embedded System? -Examples

What is an Embedded System?
-Examples

What is an Embedded System? -Examples

What is an Embedded System? -Examples

What is an Embedded System?
-Examples

What is an Embedded System?
-Fields Involved

1. Electronic Engineers;
2. Firmware Engineers;
3. STOP?!

What is an Embedded System? -Fields Involved

1. Electronic Engineers;
2. Firmware Engineers;

TILL 1980!!!

What is an Embedded System?
-Fields Involved

1. Electronic Engineers;
2. Firmware Engineers;
3. Computer Scientists;

What is an Embedded System?
-Fields Involved

1. Electronic Engineers;
2. Firmware Engineers;
3. Computer Scientists;
4. Project Managers;

What is an Embedded System?
-Fields Involved

1. Electronic Engineers;
2. Firmware Engineers;
3. Computer Scientists;
4. Project Managers;
5. Designers;

What is an Embedded System?
-Fields Involved

1. Electronic Engineers;
2. Firmware Engineers;
3. Computer Scientists;
4. Project Managers;
5. Designers;
6. Psychologists;

What is an Embedded System?
-Fields Involved

1. Electronic Engineers;
2. Firmware Engineers;
3. Computer Scientists;
4. Project Managers;
5. Designers;
6. Psychologists;
7. … WHO KNOWS!?

What is Embedded System?

-From an Article

- Business - budgeting and time management
- Software engineering/reengineering
- Teamwork
- Components reuse
- Human communication - written and spoken
- Testing and Validation
- Creating models

http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=949E14C84967916817648D29CADE6EFA?doi=10.1.1.217.7157&rep=rep1&type=pdf

What is Embedded System?

-From an Article

- **Business - budgeting and time management**
- Software engineering/reengineering
- Teamwork
- Components reuse
- Human communication - written and spoken
- Testing and Validation
- Creating models

http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=949E14C84967916817648D29CADE6EFA?doi=10.1.1.217.7157&rep=rep1&type=pdf

What is Embedded System?

-From an Article

- **Business - budgeting and time management**
- Software engineering/reengineering
- **Teamwork - with people with different backgrounds!**
- Components reuse
- Human communication - written and spoken
- Testing and Validation
- Creating models

http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=949E14C84967916817648D29CADE6EFA?doi=10.1.1.217.7157&rep=rep1&type=pdf

What is Embedded System?

-From an Article

- **Business - budgeting and time management**
- Software engineering/reengineering
- **Teamwork - with people with different backgrounds!**
- **Components reuse**
- Human communication - written and spoken
- Testing and Validation
- Creating models

http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=949E14C84967916817648D29CADE6EFA?doi=10.1.1.217.7157&rep=rep1&type=pdf

What is Embedded System?

-From an Article

- **Business - budgeting and time management**
- Software engineering/reengineering
- **Teamwork - with people with different backgrounds!**
- **Components reuse**
- **Human communication - written and spoken**
- Testing and Validation
- Creating models

http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=949E14C84967916817648D29CADE6EFA?doi=10.1.1.217.7157&rep=rep1&type=pdf

What is Embedded System?

-From an Article

- **Business - budgeting and time management**
- Software engineering/reengineering
- **Teamwork - with people with different backgrounds!**
- **Components reuse**
- **Human communication - written and spoken**
- Testing and Validation
- Creating models

http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=949E14C84967916817648D29CADE6EFA?doi=10.1.1.217.7157&rep=rep1&type=pdf

What is an Embedded System?
-Formal Definition

What is an Embedded System? -Formal Definition

Everything that is not a computer

Start from an High Level Design Diagram

- System Design (Hardware, Software, Mechanical);
- Sub-Systems Design;
- BOM (Bill of Materials);
- Costs of the entire system and parts!
- How you integrate stuffs? Why? In detail!
- Signals! Analyze the signals;
- Interfaces, «Why you used the SPI interface instead of I2C?»;
- Interface with other instruments! Smartphones, PCs, Xbox Kineckt?
- User Interface?

Start from an High Level Design Diagram

- Let’s try to do a draft of one of your projects!

Critical Systems

Critical Systems

https://www.youtube.com/watch?v=gp_D8r-2hwk

Critical Systems

2. ANALYSIS OF THE FAILURE

2.1 CHAIN OF TECHNICAL EVENTS

In general terms, the Flight Control System of the Ariane 5 is of a standard design. The attitude of the launcher and its movements in space are measured by an Inertial Reference System (SRI). It has its own internal computer, in which angles and velocities are calculated on the basis of information from a "strap-down" inertial platform, with laser gyros and accelerometers. The data from the SRI are transmitted through the databus to the On-Board Computer (OBC), which executes the flight program and controls the nozzles of the solid boosters and the Vulcain cryogenic engine, via servovalves and hydraulic actuators.

In order to improve reliability there is considerable redundancy at equipment level. There are two SRIs operating in parallel, with identical hardware and software. One SRI is active and one is in "hot" stand-by, and if the OBC detects that the active SRI has failed it immediately switches to the other one, provided that this unit is functioning properly. Likewise there are two OBCs, and a number of other units in the Flight Control System are also duplicated.

The design of the Ariane 5 SRI is practically the same as that of an SRI which is presently used on Ariane 4, particularly as regards the software.

Based on the extensive documentation and data on the Ariane 501 failure made available to the Board, the following chain of events, their inter-relations and causes have been established, starting with the destruction of the launcher and tracing back in time towards the primary cause.

- The launcher started to disintegrate at about H0 + 39 seconds because of high aerodynamic loads due to an angle of attack of more than 20 degrees that led to separation of the boosters from the main stage, in turn triggering the self-destruct system of the launcher.
- This angle of attack was caused by full nozzle deflections of the solid boosters and the Vulcain main engine.
- These nozzle deflections were commanded by the On-Board Computer (OBC) software on the basis of data transmitted by the active Inertial Reference System (SRI 2). Part of these data at that time did not contain proper flight data, but showed a diagnostic bit pattern of the computer of the SRI 2, which was interpreted as flight data.
- The reason why the active SRI 2 did not send correct attitude data was that the unit had declared a failure due to a software exception.
- The OBC could not switch to the back-up SRI 1 because that unit had already ceased to function during the previous data cycle (72 milliseconds period) for the same reason as SRI 2.
- The internal SRI software exception was caused during execution of a data conversion from 64-bit floating point to 16-bit signed integer value. The floating point number which was converted had a value greater than what could be represented by a 16-bit signed integer. This resulted in an Operand Error. The data conversion instructions (in Ada code) were not protected from causing an Operand Error, although other conversions of comparable variables in the same place in the code were protected.
- The error occurred in a part of the software that only performs alignment of the strap-down inertial platform. This software module computes meaningful results only before lift-off. As soon as the launcher lifts off, this function serves no purpose.
- The alignment function is operative for 50 seconds after starting of the Flight Mode of the SRIs which occurs at H0 - 3 seconds for Ariane 5. Consequently, when lift-off occurs, the function continues for approx. 40 seconds of flight. This time sequence is based on a requirement of Ariane 4 and is not required for Ariane 5.
- The Operand Error occurred due to an unexpected high value of an internal alignment function result called BH, Horizontal Bias, related to the horizontal velocity sensed by the platform. This value is calculated as an indicator for alignment precision over time.
- The value of BH was much higher than expected because the early part of the trajectory of Ariane 5 differs from that of Ariane 4 and results in considerably higher horizontal velocity values.

Critical System Definition

- Death or highly risk for people;
- $$$
- Environmental Dangers
- Formal Definition:

$$\textbf{Critycal Sys} \leq \frac{1}{10^9} \left[\frac{life}{hours}\right]$$

Others?

Critical Systems

Infusion Pumps:
Infusion pumps have been a source of multiple patient safety concerns, and problems with such pumps have been linked to more than 56,000 adverse event reports from 2005 to 2009, including at least 500 deaths.[1] As a result, the U.S. Food and Drug Administration(FDA) has launched a comprehensive initiative to improve their safety, called the Infusion Pump Improvement Initiative.[2] The initiative proposed stricter regulation of infusion pumps.

It cited software defects, **user interface issues**, and mechanical or electrical failures as the main causes of adverse events.

1. Pharmacy Practice News, May 2010
2.^ http://www.fda.gov/MedicalDevices/ProductsandMedicalProcedures/GeneralHospitalDevicesandSupplies/InfusionPumps/ucm205424.htm

Principles of Applied Cognitive Psychology
-
About HCI Human Computer Interaction

Classical Ergonomics

We care about:

- Operator Posture
- Anthropometric measures

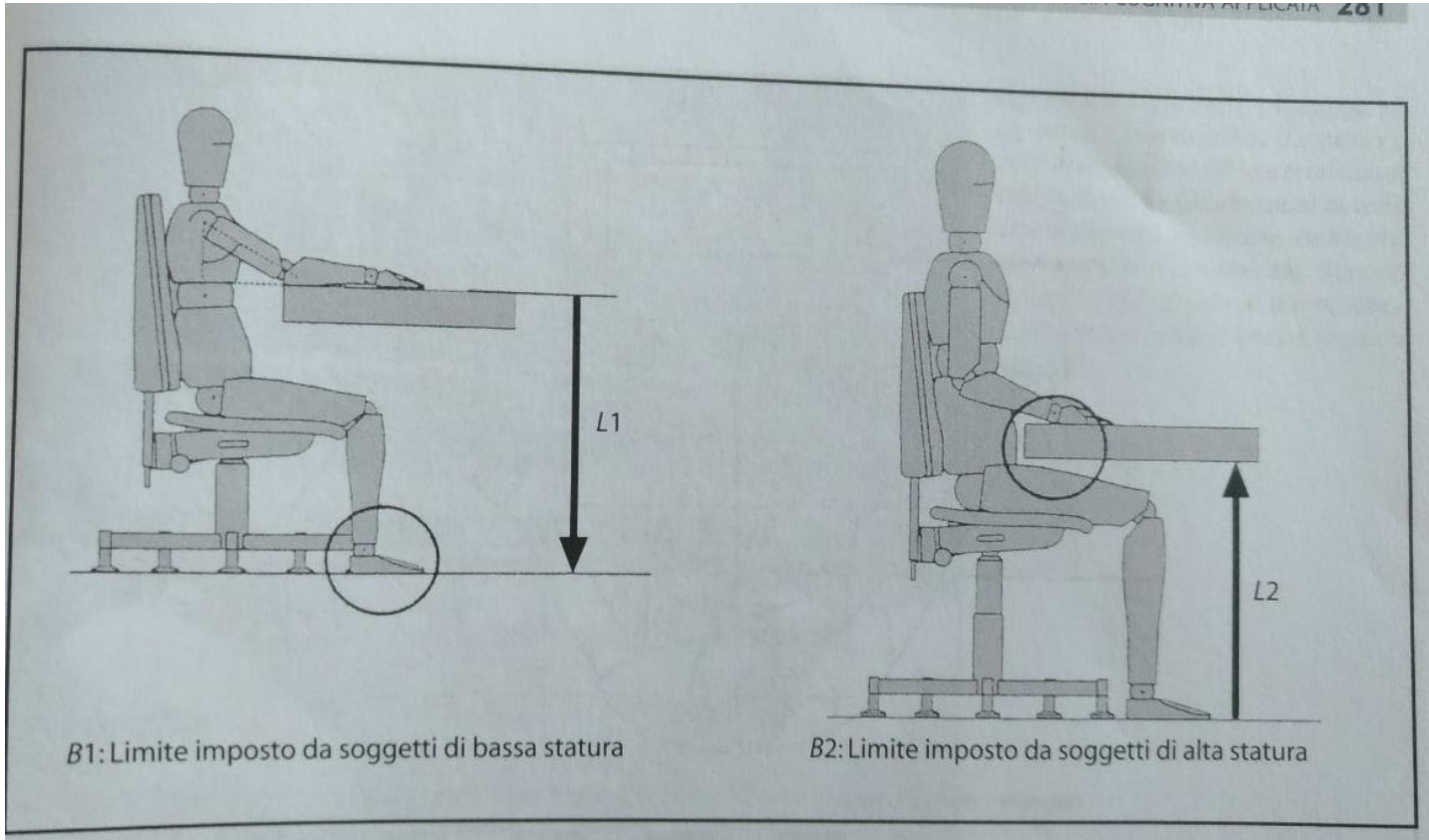


fig. 14.1. *Un esempio dello studio della postazione di lavoro in base ai parametri antropometrici dell'operatore.*

Cognitive Ergonomics

Combination of Cognitive Psychology and Ergonomics applied to understand the interface [Norman and Darper 1986]

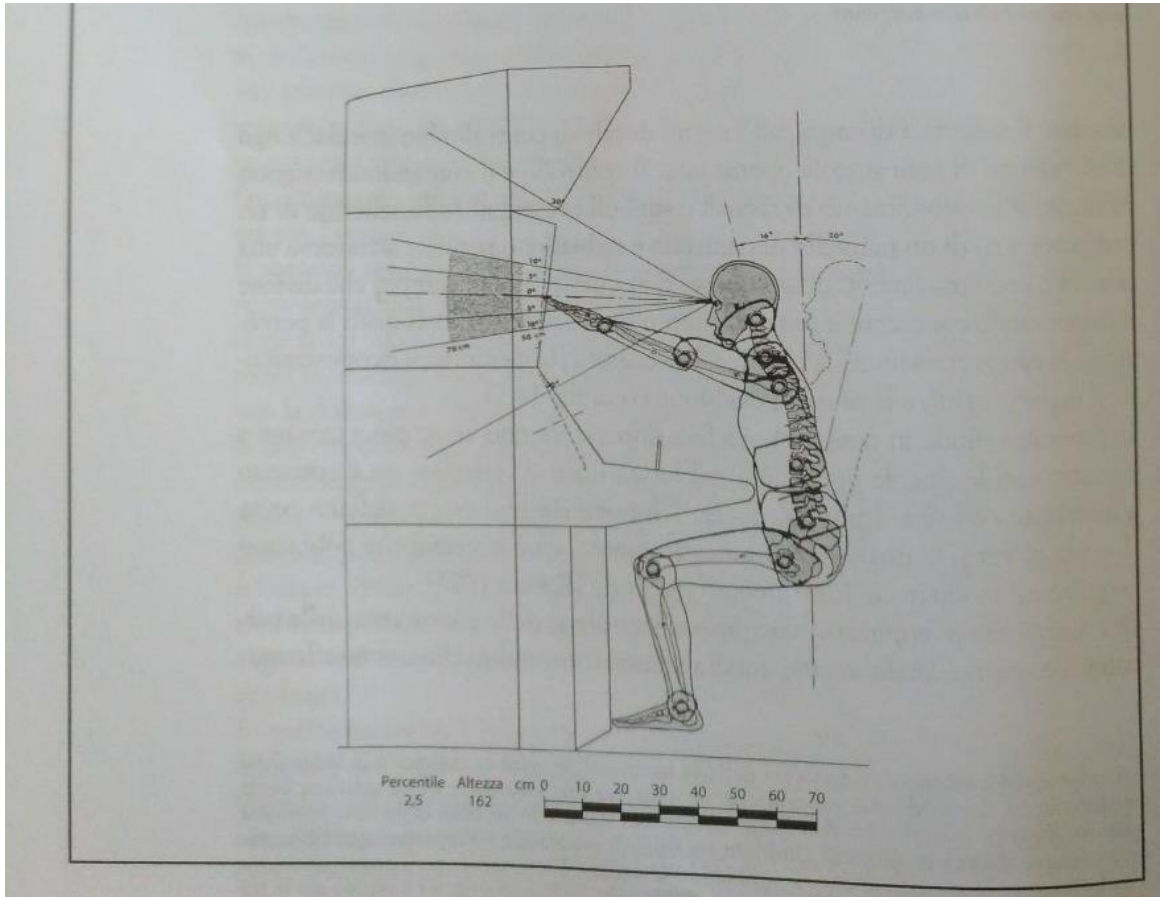


fig. 14.2. *Misurazioni e valutazioni della postazione di lavoro per le funzioni visuomotorie dell'utente.*

Interfaces

Can be defined as:
The way about we do something with an instrument.
Actions we have to do and the way about the instrument respond.

The main areas of intervention of cognitive psychology

- Design everyday use objects
- Creation and large-scale production of PCs
- Interface Design

Designing Everyday Use Objects

Biederman [1] in 1987 calculated that in the environment in which we live there are:

30.000 recognizable objects, let's say 20.000 to be VERY optimist (scissors, glasses, cars, etc.).

Hypothesis:

1 min per object = 20.000 mins = 8 working weeks

(40 hours / week)

[1]Joseph Biederman is Chief of the Clinical and Research Programs in Pediatric Psychopharmacology and Adult ADHD at the Massachusetts General Hospital, professor of psychiatry at Harvard Medical School. Biederman is Board Certified in General and Child Psychiatry.

Designing Everyday Use Objects

Biederman [1] in 1987 calculated that in the environment in which we live there are:

30.000 recognizable objects, let's say 20.000 to be VERY optimist (scissors, glasses, cars, etc.).

Hypothesis:

1 min per object = 20.000 mins =

8 working weeks (40 hours / week)

Solution?

[1]Joseph Biederman is Chief of the Clinical and Research Programs in Pediatric Psychopharmacology and Adult ADHD at the Massachusetts General Hospital, professor of psychiatry at Harvard Medical School. Biederman is Board Certified in General and Child Psychiatry.

Solution:

You should involve:

1. Affordance
2. Mapping
3. Feedback

Affordance

3 factors to involve:

- Mode of operation of mental processes
- **Information contained in the aspect that objects have**
- **The designer's ability to make objects clear and understandable**

Authorization (permitted operations) for
example:

- Balls (do you want to kick them?)

Affordance

3 factors to involve:

- Mode of operation of mental processes
- **Information contained in the aspect that objects have**
- **The designer's ability to make objects clear and understandable**

Authorization (permitted operations) for example:

- Balls (do you want to kick it?)

Affordance

3 factors to involve:

- Mode of operation of mental processes
- **Information contained in the aspect that objects have**
- **The designer's ability to make objects clear and understandable**

Authorization (permitted operations) for example:

- Balls (do you want to kick it?)

Affordance

3 factors to involve:

- Mode of operation of mental processes
- **Information contained in the aspect that objects have**
- **The designer's ability to make objects clear and understandable**

Authorization (permitted operations) for example:

- Balls (do you want to kick it?)

Affordance

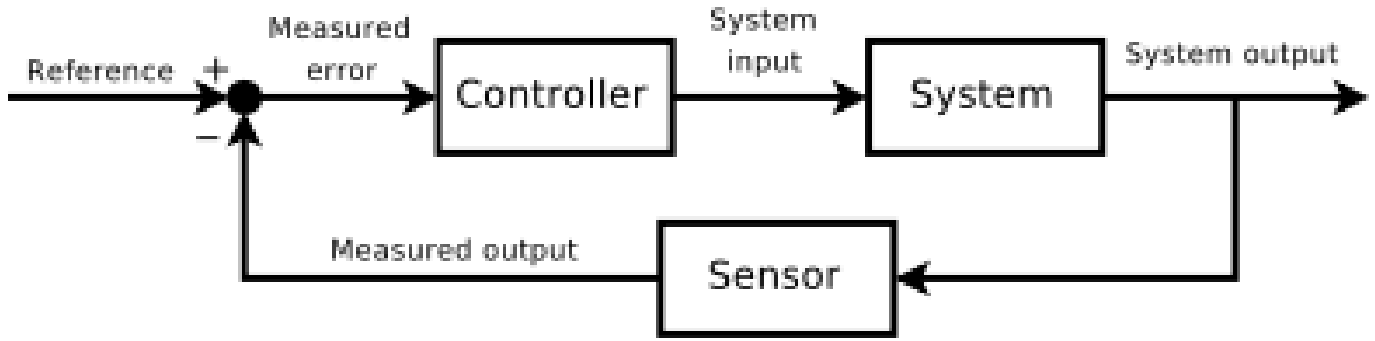
To have a good affordance we have to aim to a good Design

Good Affordance = Good Design

A Good Design

Norman [1] says that a Good Design is achieved when some general principles are satisfied:

- Use of a good conceptual model, in this way we clarify to the user the functional characteristics (what does this button do?)
- Follow a Good **mapping**
- Apply the **feedback principle**



[1] Donald Arthur Norman (25 dec 1935) American Engineer.

Conceptual Model Violations

Some Examples?

Conceptual Model Violations

Some Examples?

Set the time of this clock!

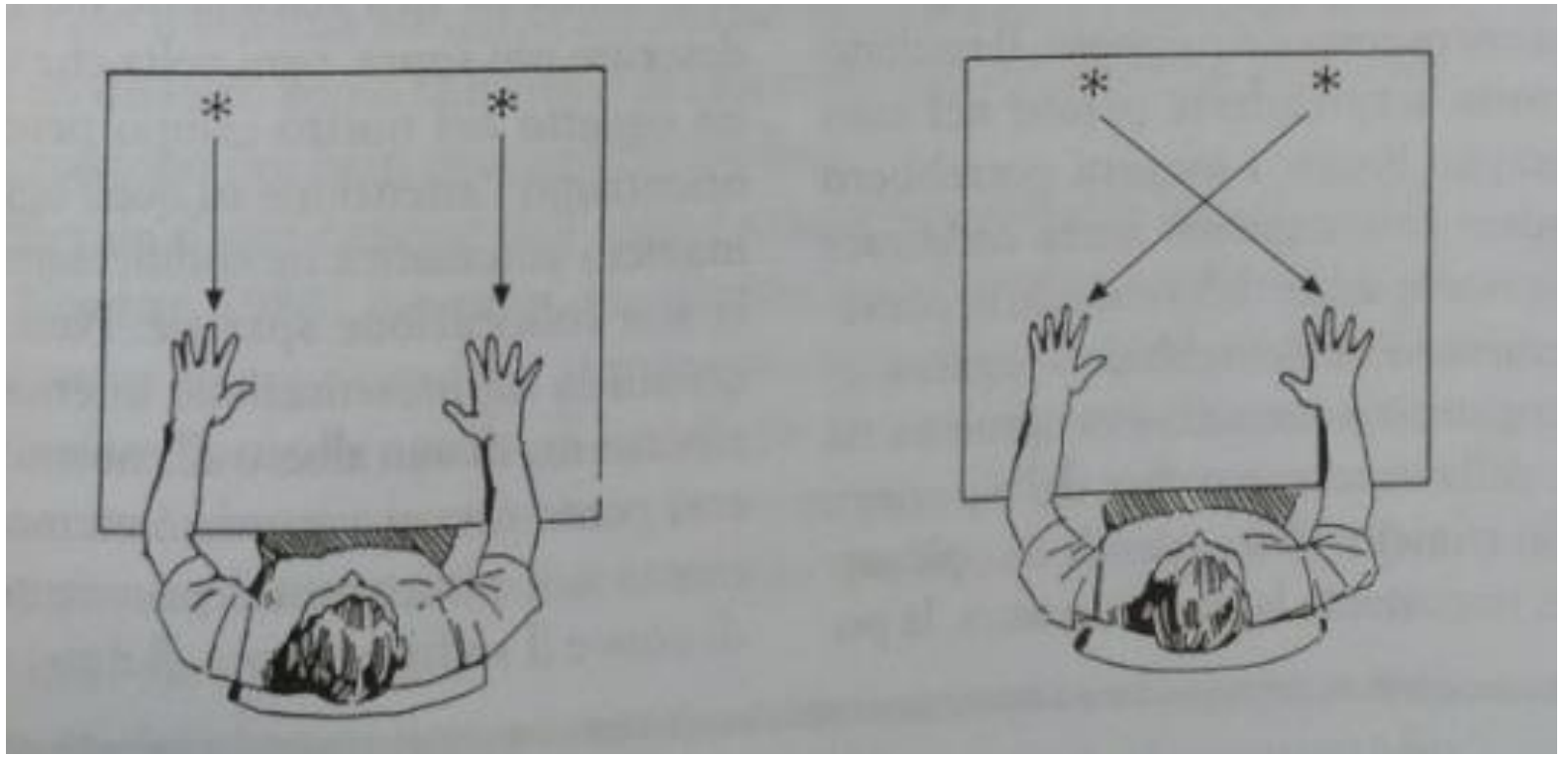
Conceptual Model NON Violations

Some Examples?

Set the time of this clock!

Mapping

Following a good mapping means: Clear relationships between actions and results, in particular between commands and effects.



Mapping’s Violation

You can see it everyday when you want to cook something…

Mapping’s Violation

Relation between Fires and Knobs of your Hob.

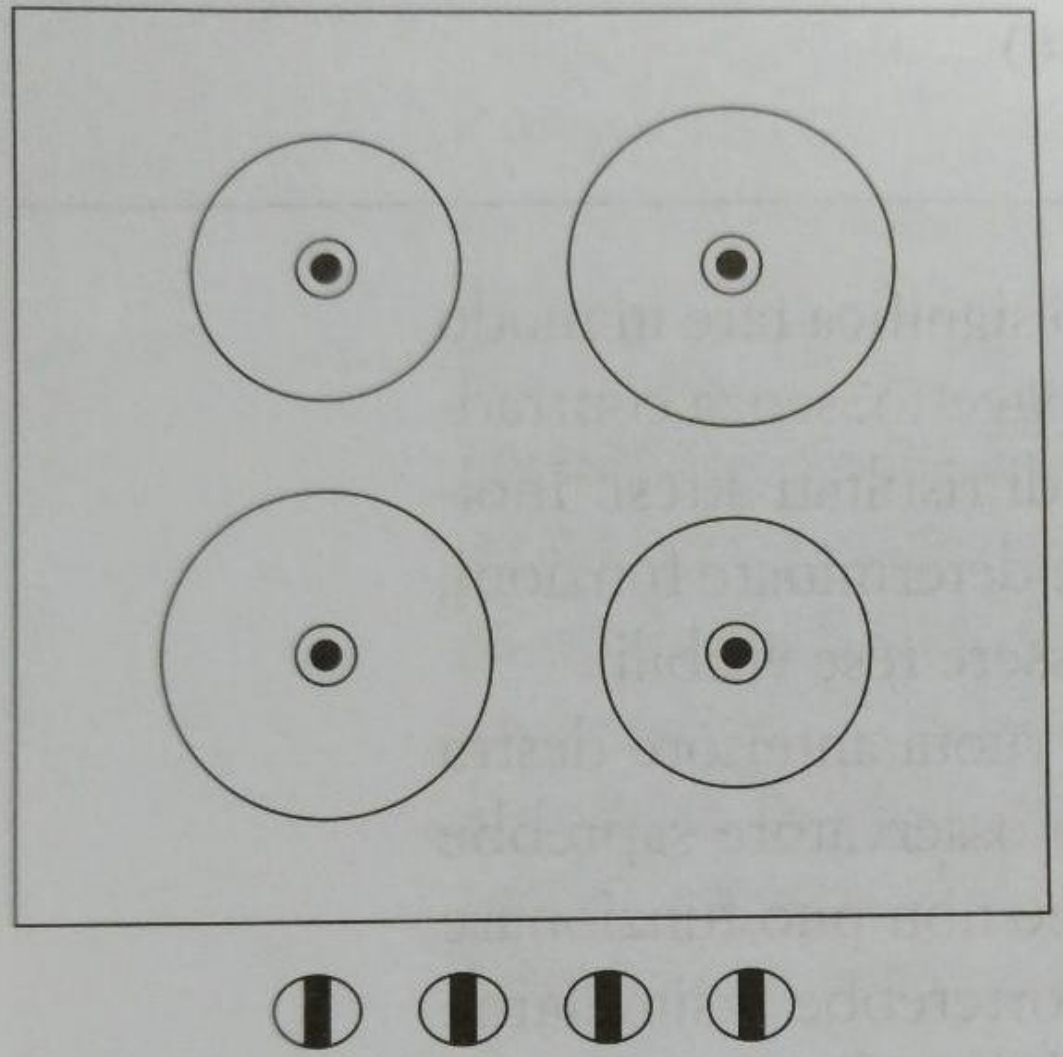
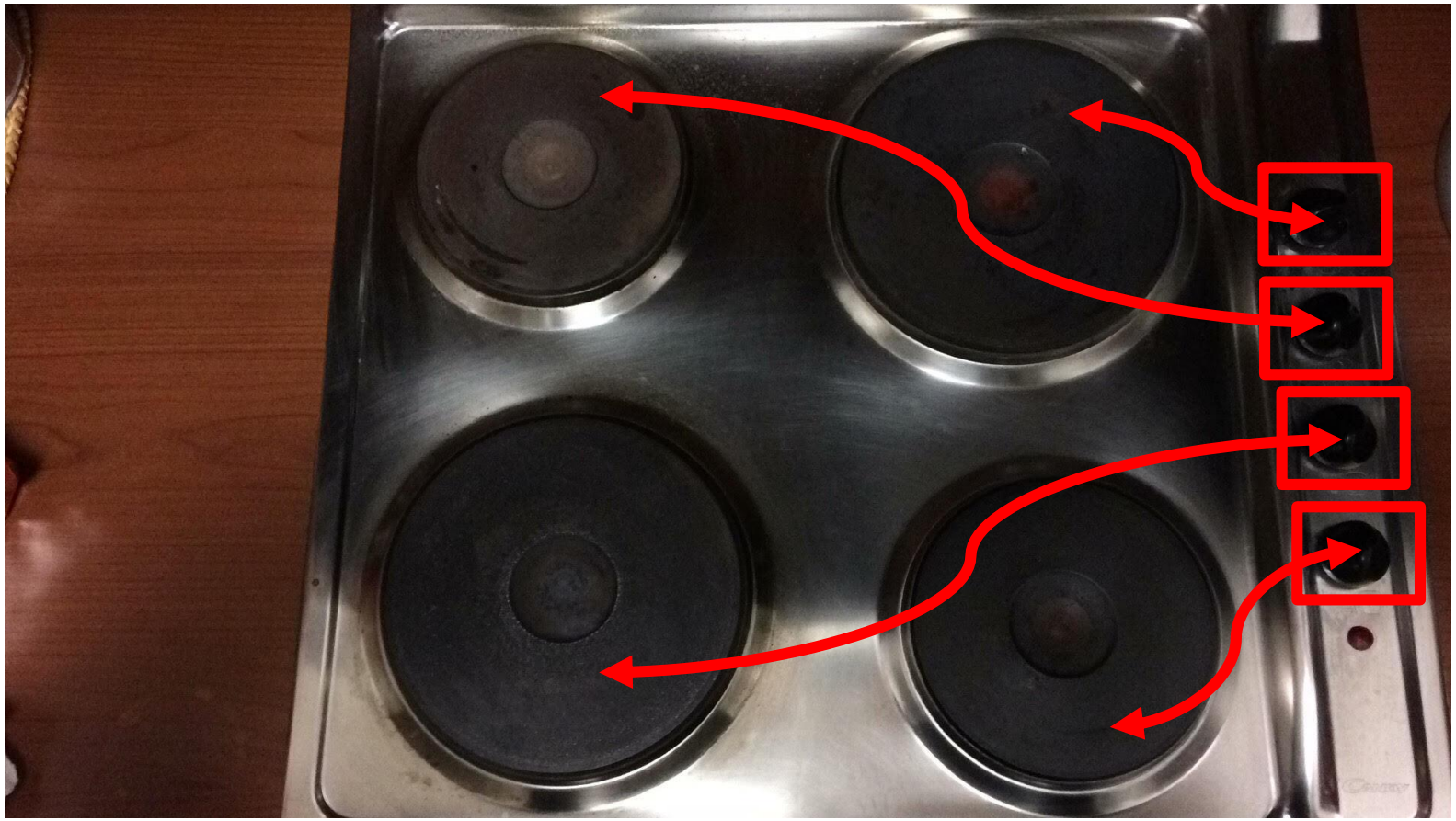


fig. 14.4. *Tradizionale fornello a 4 fuochi disposti a quadrato. Quale sarà il «mapping» con le quattro manopole di accensione e spegnimento?*

Fonte: Norman [1988].

Which knob turn on which Fire?
-Try to guess

Which knob turn on which Fire?



A Good Mapping

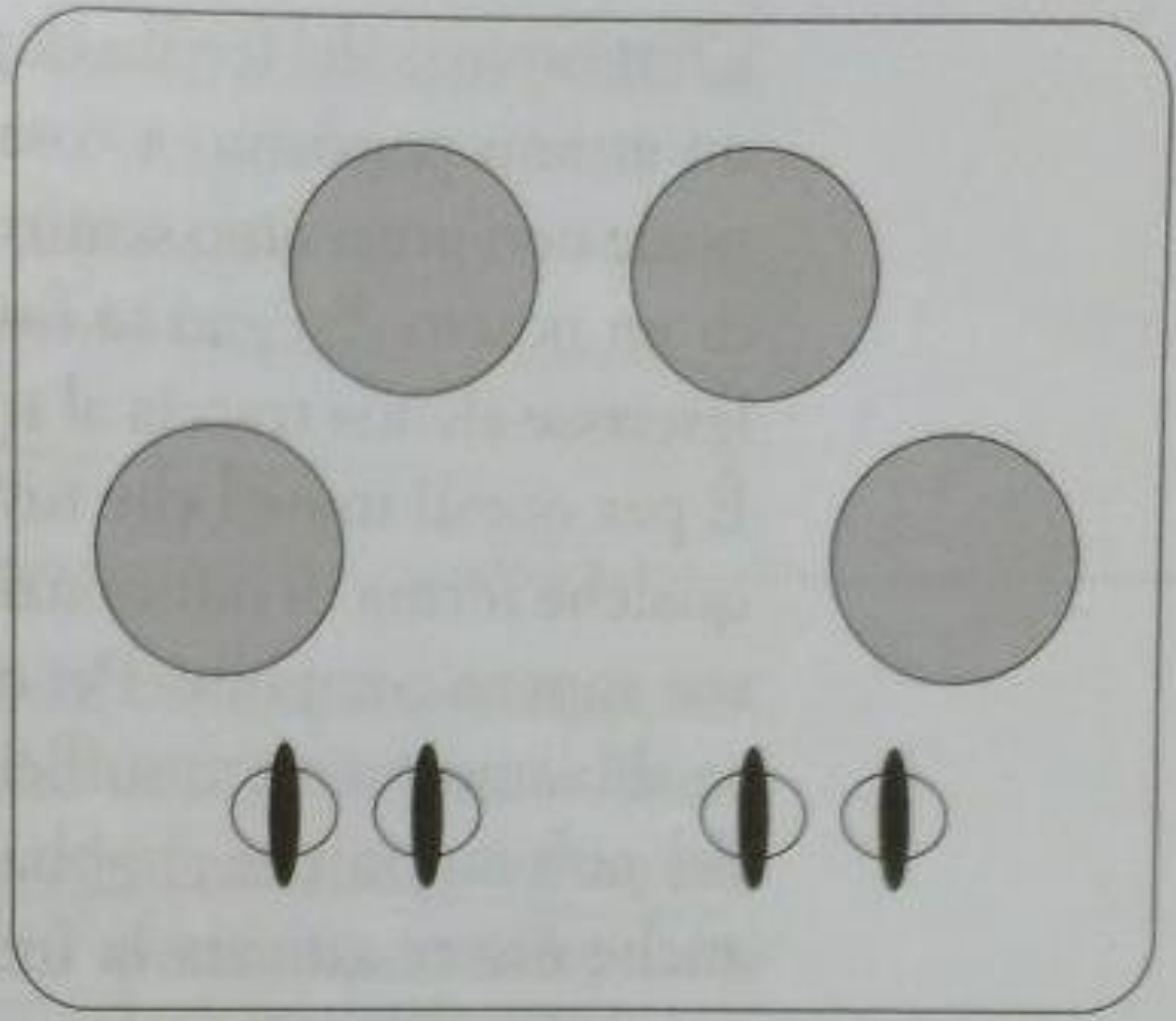
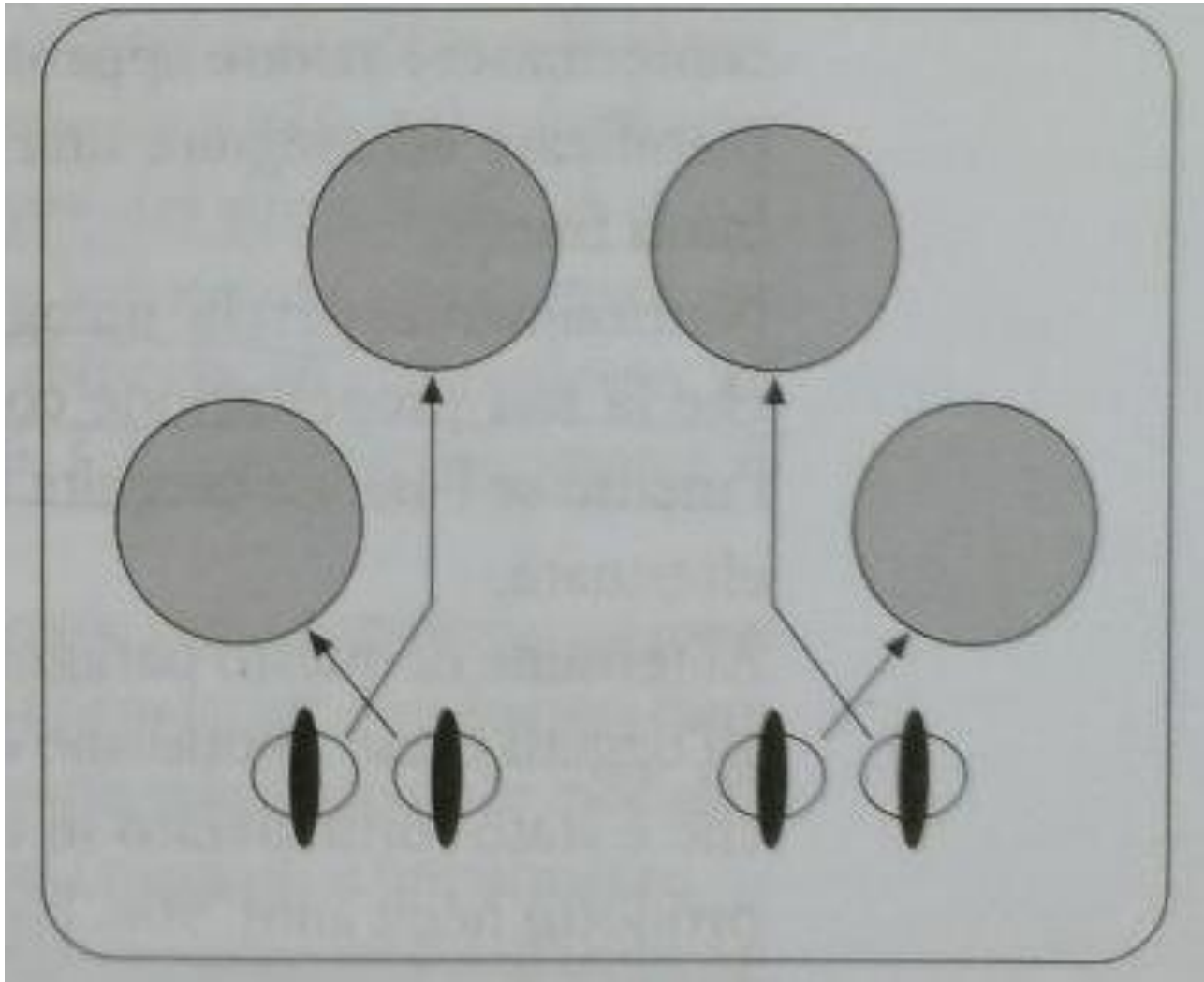


fig. 14.6. *Piano di cottura con 4 fuochi e 4 manopole. Il «mapping» tra fuochi e manopole appare di ottima usabilità.*

Fonte: Arielli [2003].

A (ruined) Good Mapping

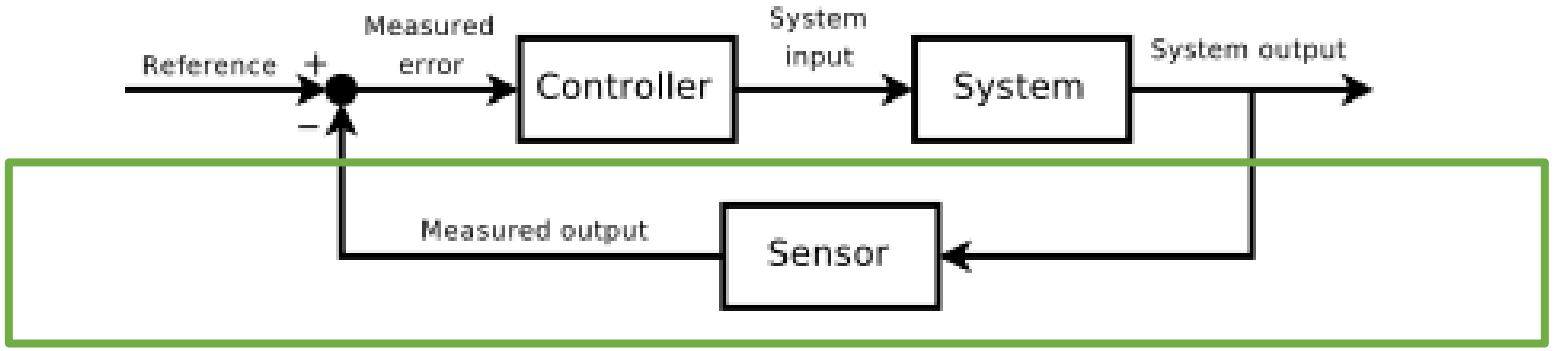


Feedback Principle

Applying the principle of feedback means providing a message that includes a sense of the user, using this sense the user receives retroactive information.

Feedback Principle

Applying the principle of feedback means providing a message that includes a sense of the user, using this sense the user receives retroactive information.



Feedback Principle

Applying the principle of feedback means providing a message that includes a sense of the user, using this sense the user receives retroactive information.

Principio di feedback

Applicare il principio di feedback significa prevedere un messaggio che includa un senso con il quale l’utente riceva un informazione retroattiva.

For a good level of usability it is necessary to provide at least one feedback.

Product Prototyping and Service Definition

What does Product/Service prototyping means?

Models – Delivering a Prototype

Waterfall model

Agile Methodology

- Waterfall VS Agile
- Scrum and Kanban

What is an Embedded System?

Principles of Applied Cognitive Psychology - About HCI Human Computer Interaction

- Affordance
- Feedback
- Mapping

Examples on students projects