
FII018: INGEGNERIA DEL SOFTWARE

Non Functional Requirements

Lecturer: Prof. Henry Muccini
Università degli Studi dell'Aquila



Dipartimento di Ingegneria e Scienze
dell'Informazione e Matematica

Università degli Studi dell'Aquila

asd

Copyright Notice

The material in these slides may be freely reproduced and distributed, partially or totally, as far as an explicit reference or acknowledge to the material author is preserved.

Some of the slides presented in this lecture comes from the textbook, some others from Chapter 4 of the Sommerville software engineering book

Henry Muccini

asd

Text Book Reading

Preface+Chapter 1+Chapter 2+Chapter 4

@ «Object-Oriented Software Engineering – using UML, Patterns, and Java», by Bernd Bruegge and Allen H. Dutoit

Mandatory Reading

"Fundamentals of Software Engineering" Carlo Ghezzi, Mehdi Jazayeri, and Dino Mandrioli, chapter 3



Mandatory Reading

Chapter 4 from
«Software Engineering 10th Edition», by
Ian Sommerville
<http://iansommerville.com/software-engineering-book/>



asd

What is a requirement?

- high-level abstract **statement of a service** or of a system **constraint**
- detailed **mathematical** functional specification.
- define the system **boundaries**

What is inside,
what is outside the
system?

asd

Functional and non-functional requirements

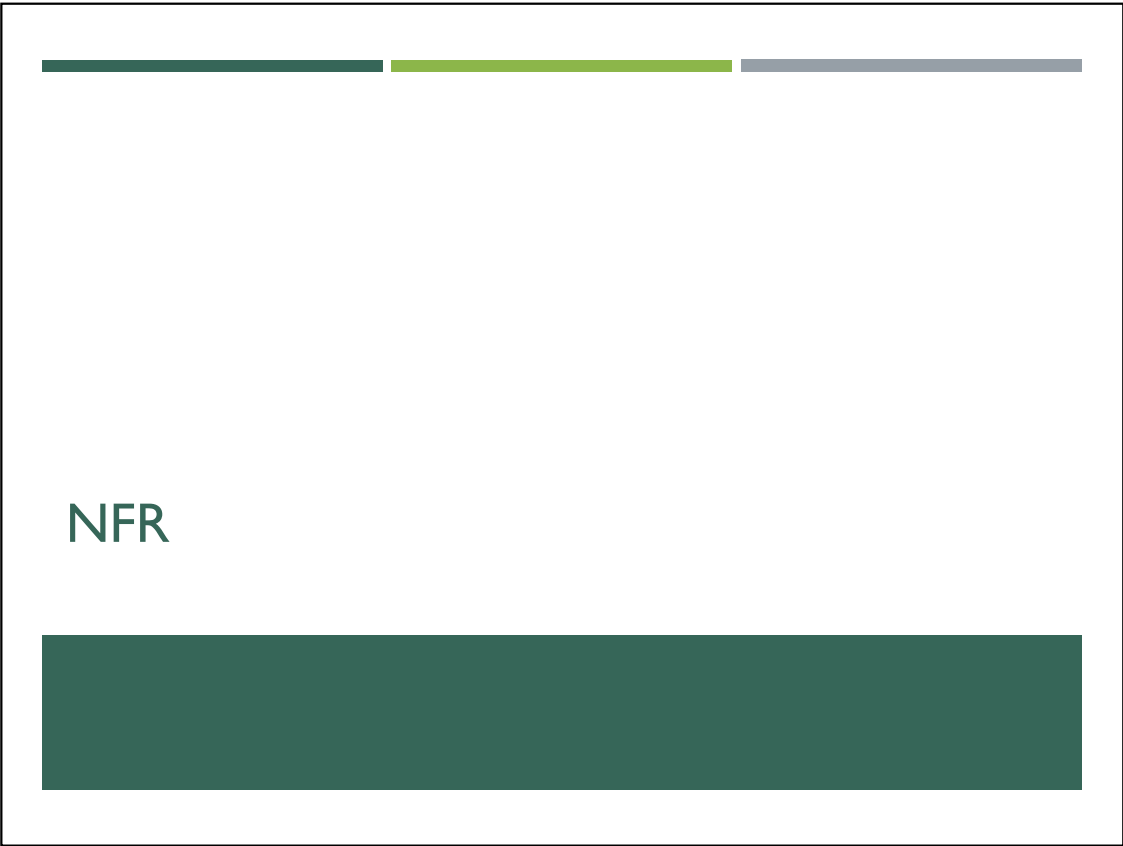
Functional requirements

- Statements of **services** the system should provide, how the system should react to particular inputs and how the system should behave in particular situations.
- May state what the system should not do.

Non-functional requirements

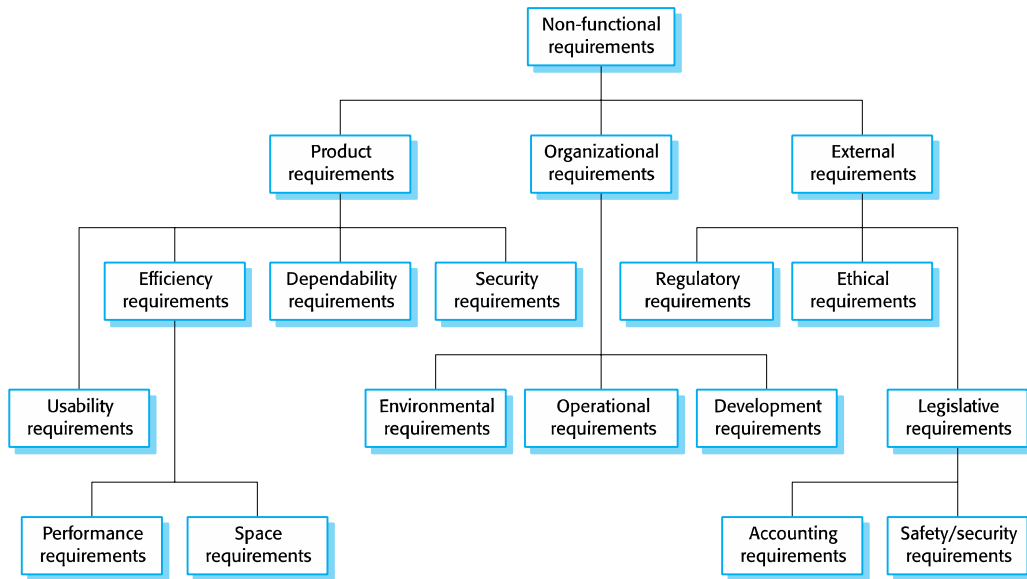
- **Constraints** on the services or functions offered by the system such as timing constraints, constraints on the development process, standards, etc.
- Often **apply to the system as a whole** rather than individual features or services.

asd



asd

Types of nonfunctional requirement



asd

Non-functional classifications

Product requirements

- Requirements which specify that the **delivered product must behave in a particular** way e.g. execution speed, reliability, etc.

Organisational requirements

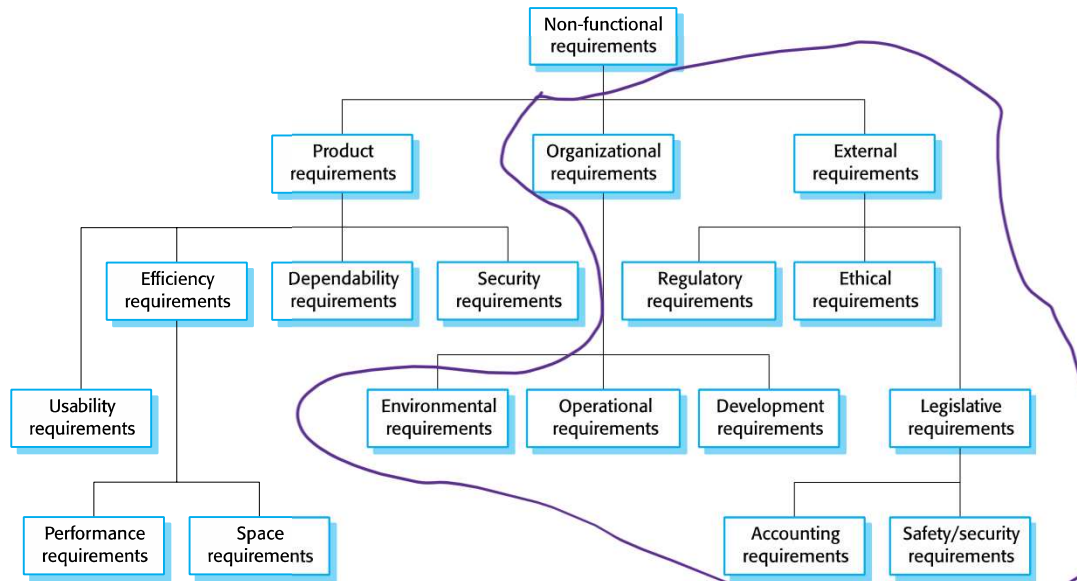
- Requirements which are a consequence of **organisational policies and procedures** e.g. process standards used, implementation requirements, etc.

External requirements

- Requirements which arise from factors which are **external to the system and its development process** e.g. interoperability requirements, legislative requirements, etc.

asd

Types of nonfunctional requirement



asd

Dashboard di Monitoraggio e Gestione delle Alluvioni

Obiettivo:

Realizzare una dashboard che permetta di monitorare in tempo reale dati provenienti da sensori distribuiti in zone a rischio alluvione all'interno di una città o di una regione. Questi sensori rileveranno parametri chiave come livello dell'acqua, intensità della pioggia, saturazione del suolo e velocità del vento. Ogni sensore avrà un codice identificativo univoco e sarà associato a una zona specifica soggetta a rischio idrogeologico.

Descrizione del sistema:

I sensori, distribuiti in diverse aree geografiche come fiumi, canali, bacini di raccolta e aree urbane a rischio, raccoglieranno periodicamente i seguenti parametri:

- **Livello dell'acqua**
- **Velocità di flusso dei fiumi**
- **Pioggia cumulativa**
- **Saturazione del terreno**
- **Velocità e direzione del vento**

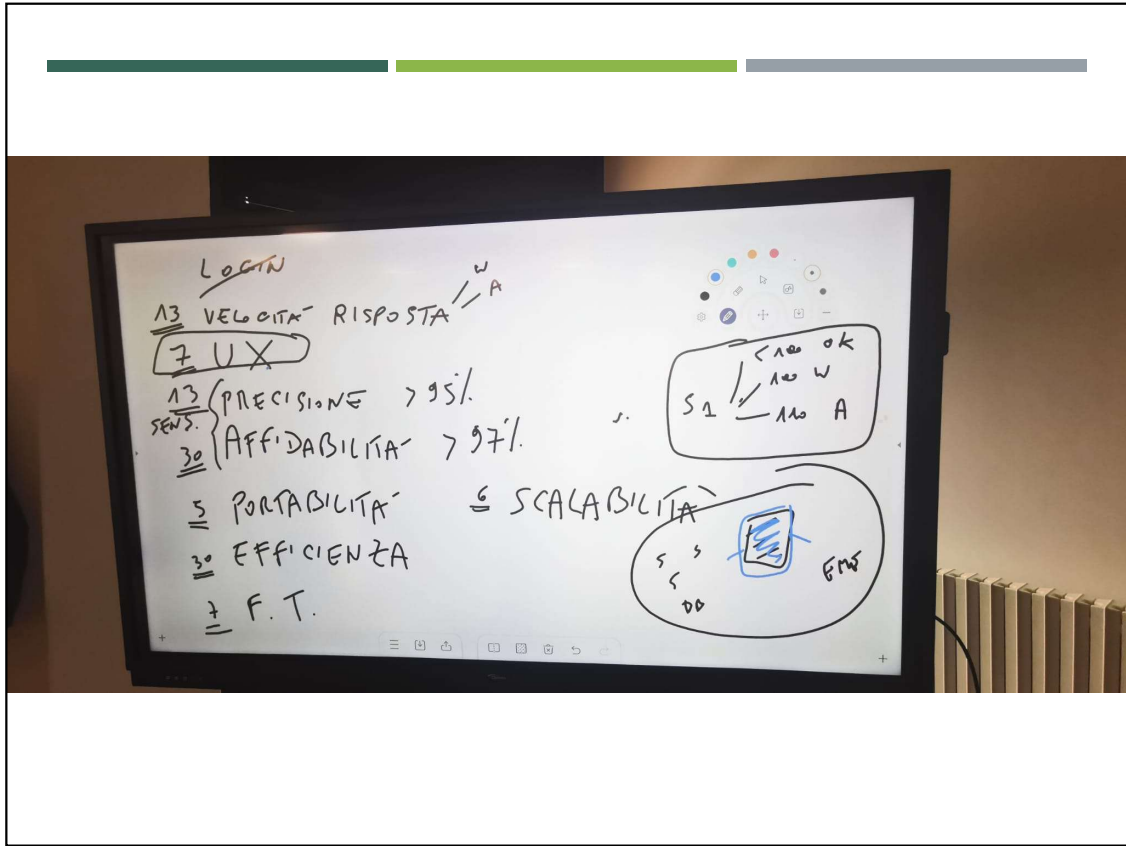
Funzionalità base:

- Visualizzazione dei dati
- Possibilità di selezionare un'area geografica specifica e monitorare i sensori associati a tale area.
- Notifica di malfunzionamento dei sensori
- Invio di allarmi prioritari in caso di pericolo imminente, come il rischio di esondazione in corso.

Ogni sensore invierà periodicamente i dati al sistema centrale insieme allo stato di funzionamento del sensore stesso (0-1). La dashboard mostrerà i dati raccolti e informerà i gestori di eventuali superamenti delle soglie critiche predefinite, come ad esempio il superamento del livello di allerta di un fiume o l'intensità delle piogge.

IN CLASSE

asd



asd



Progetto 2: *La tua spesa, il nostro click!* (1/2)

Ambito d'azione: welfare

Contesto: singolo borgo (per iniziare)

Descrizione: Per chi vive in un borgo, anche le attività più semplici possono risultare complicate. L'obiettivo di questo progetto è aiutare le persone più anziane residenti a sviluppare autonomia e fiducia nei confronti dei soci della cooperativa. Da quanto ci è stato raccontato, una delle problematiche dei piccoli borghi è che spesso non vi sono presenti negozi di alimentari. In questo senso, i soci della cooperativa si preoccupano non solo di richiedere a chi non ha modo di spostarsi dal borgo se c'è necessità di qualcosa, ma anche poi di consegnarglielo non appena possibile. A volte questa cosa può creare un po' di riluttanza da parte degli anziani, che possono sentirsi a disagio nel chiedere tale aiuto. In questo contesto, lo scopo di questo progetto è la creazione di un software per la gestione della spesa alimentare (e non) nel borgo a partire da un set predefinito e ampliabile di prodotti di prima necessità (pane, latte, uova, farina). Gli anziani del paese potranno richiedere ciò di cui hanno bisogno nei seguenti modi:

IN CLASSE

asd

Progetto 3: Viaggi cooperativi

Descrizione: I borghi sono posti meravigliosi. Una delle loro caratteristiche è la lontananza dalle città. Questa loro affascinante peculiarità si scontra però con la praticità nello svolgere anche le più semplici attività. Inoltre spesso questi posti sono abitati da persone anziane e non automunite e i servizi pubblici non sono presenti. L'obiettivo di questo progetto è quello di fornire uno strumento software per permettere ad abitanti dello stesso borgo, o di borghi vicini, di mettersi in contatto in modo da ottimizzare i viaggi verso la città. In questo senso, chi deve andare in città ed è automunito, pubblicherà sull'applicazione il suo evento con orario di partenza e di ritorno, nonché la disponibilità a trasportare persone dei borghi vicini. Allo stesso tempo, chi non è munito di mezzi di trasporto potrà inserire un annuncio di richiesta di passaggio indicando la fascia oraria nella quale deve permanere in città. Le richieste che non potranno essere soddisfatte saranno prese in carico dai soci della cooperativa o da volontari. Anche in questo caso gli anziani potranno usufruire del servizio in due modi differenti:

1. Recandosi in uno dei punti di riferimento/aggregazione del borgo comunicando le proprie necessità all'operatore preposto.
2. Chiamando uno dei soci della cooperativa o volontario il quale i) pubblica l'annuncio sull'applicazione ii) prenota la condivisione dell'auto al posto dell'anziano iii) una volta effettuata la prenotazione, avverte l'anziano.

IN CLASSE

asd

- Non-functional requirements **may be more critical than functional requirements**. If these are not met, the system **may be useless**.
- A single non-functional requirement, such as a security requirement, may **impact a number of related functional requirements**

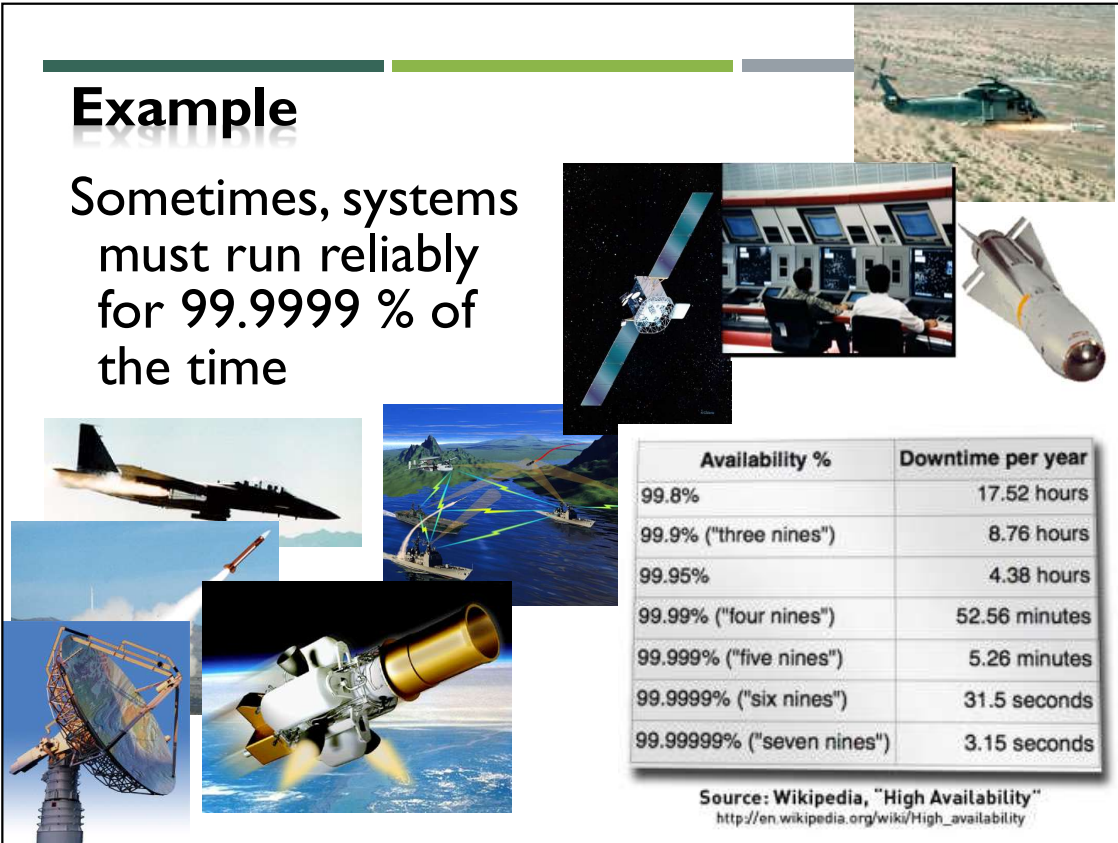
IMP: NFR may have direct or indirect relationships among them

IMP: the design/cost of a system may change dramatically when modifying the required level of NFR

asd

Example

Sometimes, systems must run reliably for 99.9999 % of the time



asd

L'evoluzione
dell'informatica
nelle attività spaziali

Elena Grifoni Winters
L'Aquila, 21 Ottobre 2024



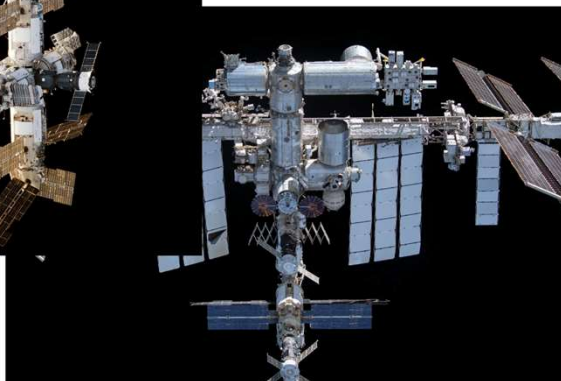
1961

Piu' software che hardware

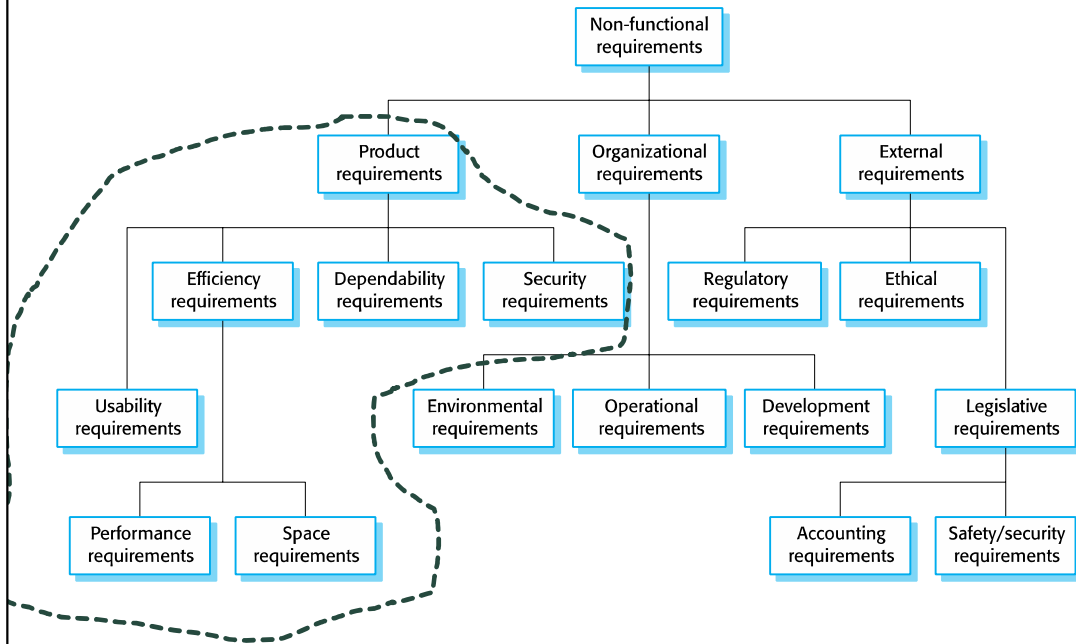


1986-2000

2024



Types of nonfunctional requirements



asd

Esercizio per casa:
selezionate e studiate in dettaglio tre NFR,
definite le relazioni tra di essi, ed
applicateli ad una delle app descritte in
precedenza

Standard ISO 25000 (<http://iso25000.com/>)

* ISO/IEC 25010: <https://iso25000.com/index.php/en/iso-25000-standards/iso-25010>
(Software Product Quality)

* ISO/IEC 25012: <http://iso25000.com/index.php/en/iso-25000-standards/iso-25012>
(Data Quality)

https://en.wikipedia.org/wiki/Non-functional_requirement

<https://www.requirements.com/Glossary/NonFunctionalRequirements/tabid/91/Default.aspx>

Textbook

asd

ISO/IEC 25010 - Modelli di qualità

SOFTWARE PRODUCT QUALITY								
FUNCTIONAL SUITABILITY	PERFORMANCE EFFICIENCY	COMPATIBILITY	INTERACTION CAPABILITY	RELIABILITY	SECURITY	MAINTAINABILITY	Flexibility	SAFETY
FUNCTIONAL COMPLETENESS FUNCTIONAL CORRECTNESS FUNCTIONAL APPROPRIATENESS	TIME BEHAVIOUR RESOURCE UTILIZATION CAPACITY	CO-EXISTENCE INTEROPERABILITY	APPROPRIATENESS RECOGNIZABILITY LEARNABILITY OPERABILITY USER ERROR PROTECTION USER ENGAGEMENT INCLUSIVITY USER ASSISTANCE SELF- DESCRIPTIVENESS	FAULTLESSNESS AVAILABILITY FAULT TOLERANCE RECOVERABILITY	CONFIDENTIALITY INTEGRITY NON-REPUDIATION ACCOUNTABILITY AUTHENTICITY RESISTANCE	MODULARITY REUSABILITY ANALYSABILITY MODIFIABILITY TESTABILITY	ADAPTABILITY SCALABILITY INSTALLABILITY REPLACEABILITY	OPERATIONAL CONSTRAINT RISK IDENTIFICATION FAIL SAFE HAZARD WARNING SAFE INTEGRATION

iso25000.com

QUALITY IN USE				
EFFECTIVENESS	EFFICIENCY	SATISFACTION	FREEDOM FROM RISK	CONTEXT COVERAGE
EFFECTIVENESS	EFFICIENCY	USEFULNESS TRUST PLEASURE COMFORT	ECONOMIC RISK MITIGATION HEALTH AND SAFETY RISK MITIGATION ENVIRONMENTAL RISK MITIGATION	CONTEXT COMPLETENESS FLEXIBILITY

asd



PRODUCT NON FUNCTIONAL REQUIREMENTS



asd

Performance

Performance engineering or SPE (Systems Performance Engineering or Software Performance Engineering) within systems engineering, encompasses the set of roles, skills, activities, practices, tools, and deliverables applied at every phase of the systems development life cycle which ensures that a solution will be designed, implemented, and operationally supported to meet the non-functional requirements for performance (such as throughput, and latency).

asd



Time to complete of a task

- ✓ the speed with which some specific workload can be completed

In general terms, **throughput** is the rate of production or the rate at which something can be processed.

- ✓ how many units of information a system can process in a given amount of time.

Latency is a time interval between the stimulation and response

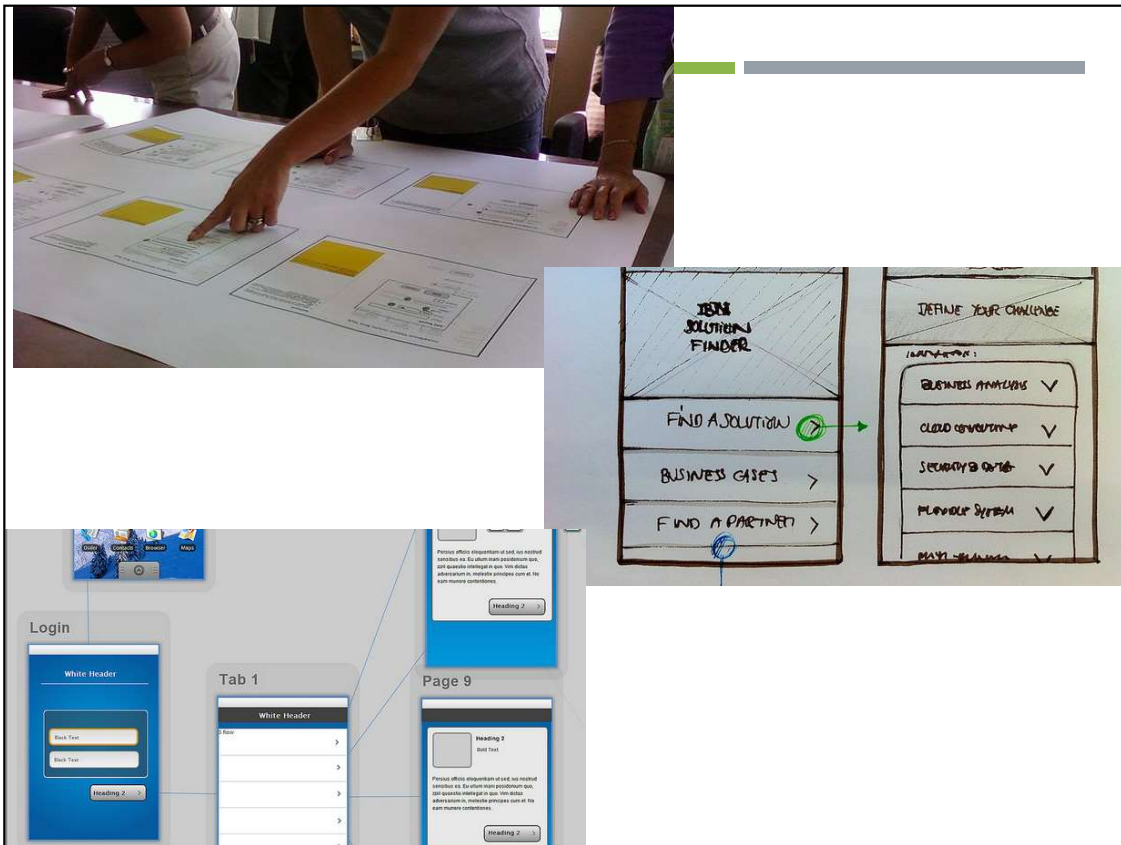
asd

Some Quality Requirements Definitions

Usability

- The ease with which actors can use a system to perform a function
- Usability is one of the most frequently misused terms (“The system is easy to use”)
- **Usability** must be measurable, otherwise it is **marketing**
 - Example: Specification of the number of steps – the measure!
 - to perform a internet-based purchase with a web browser

asd



asd



asd

Customer Reviews

Horrible upgrade!

by Agent772

This newer version is not user friendly at all. Half my screen is cropped, blurry text, and crashes often. The app and website upgrade needs to be redone, this is just ridiculous! Old app was better!

Ugly, bloated

by Richard from Seattle

Latest update is a step backwards. Interface is cluttered, graphics are blurry, almost has too much functionality. It's like they crammed the entire website into the app.

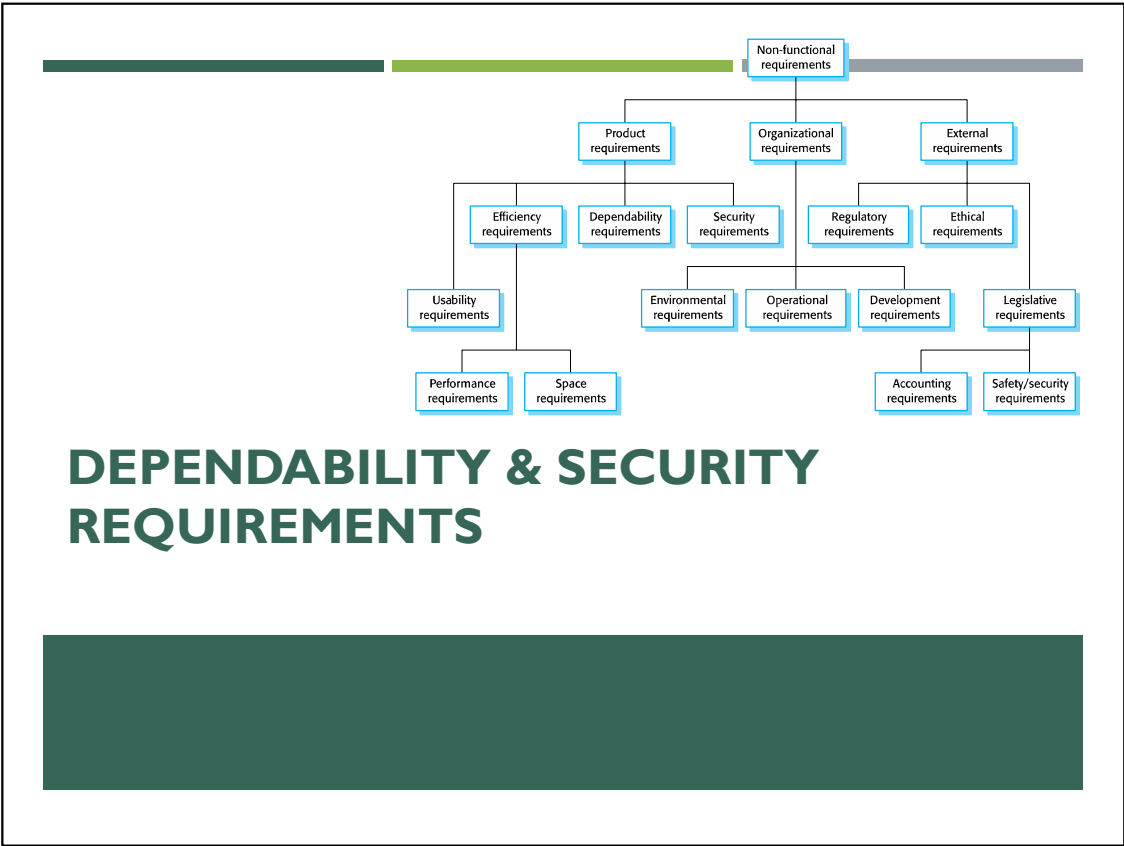
Huge step backwards

by Long Time Strategi User

This version is unusable. It's just a mobile version of the new website, which is also unusable.

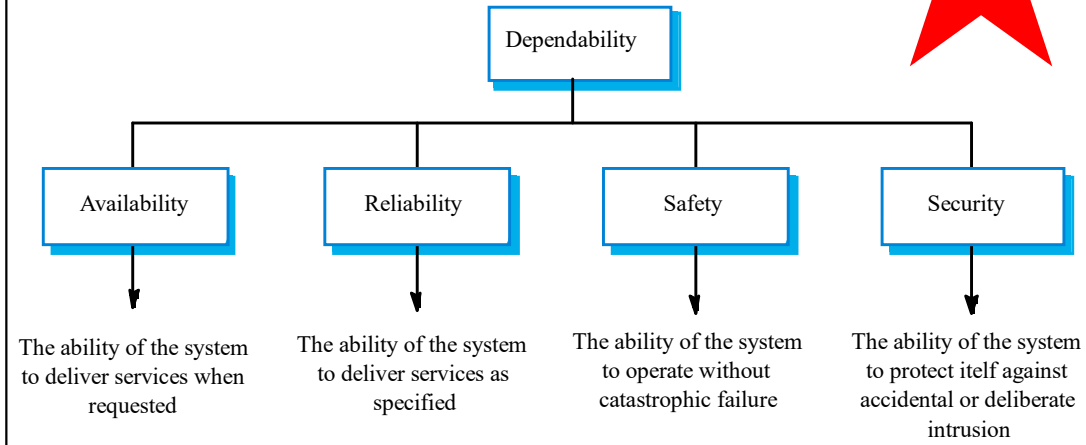
Previous version was quick and convenient to use.
Very disappointing.

asd



asd

Dimensions of dependability



asd

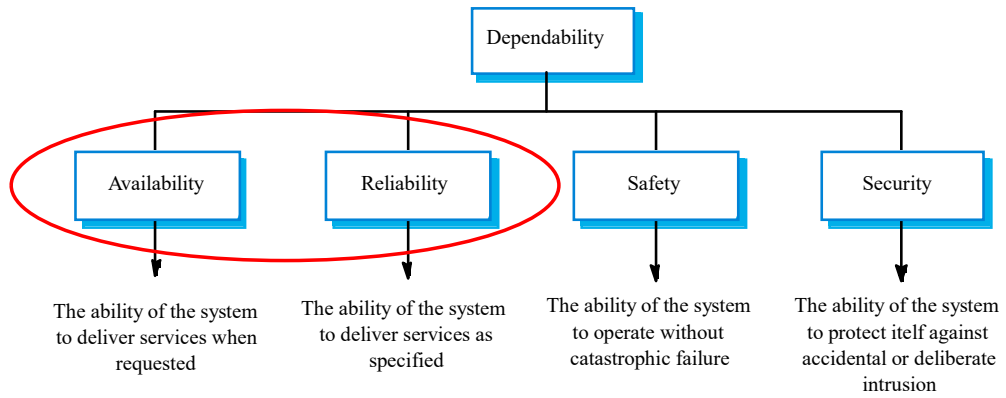
System dependability

The dependability of a system reflects the user's degree of **trust in that system.**

It reflects the extent of the **user's confidence** that it will **operate as users expect** and that it **will not 'fail' in normal use.**

asd

Dimensions of dependability



asd

Dashboard di Monitoraggio e Gestione delle Alluvioni

Obiettivo:

Realizzare una dashboard che permetta di monitorare in tempo reale dati provenienti da sensori distribuiti in zone a rischio alluvione all'interno di una città o di una regione. Questi sensori rileveranno parametri chiave come livello dell'acqua, intensità della pioggia, saturazione del suolo e velocità del vento. Ogni sensore avrà un codice identificativo univoco e sarà associato a una zona specifica soggetta a rischio idrogeologico.

Descrizione del sistema:

I sensori, distribuiti in diverse aree geografiche come fiumi, canali, bacini di raccolta e aree urbane a rischio, raccoglieranno periodicamente i seguenti parametri:

- **Livello dell'acqua**
- **Velocità di flusso dei fiumi**
- **Pioggia cumulativa**
- **Saturazione del terreno**
- **Velocità e direzione del vento**

Funzionalità base:

- Visualizzazione dei dati
- Possibilità di selezionare un'area geografica specifica e monitorare i sensori associati a tale area.
- Notifica di malfunzionamento dei sensori
- Invio di allarmi prioritari in caso di pericolo imminente, come il rischio di esondazione in corso.

Ogni sensore invierà periodicamente i dati al sistema centrale insieme allo stato di funzionamento del sensore stesso (0-1). La dashboard mostrerà i dati raccolti e informerà i gestori di eventuali superamenti delle soglie critiche predefinite, come ad esempio il superamento del livello di allerta di un fiume o l'intensità delle piogge.

IN CLASSE

asd



Progetto 2: *La tua spesa, il nostro click!* (1/2)

Ambito d'azione: welfare

Contesto: singolo borgo (per iniziare)

Descrizione: Per chi vive in un borgo, anche le attività più semplici possono risultare complicate. L'obiettivo di questo progetto è aiutare le persone più anziane residenti a sviluppare autonomia e fiducia nei confronti dei soci della cooperativa. Da quanto ci è stato raccontato, una delle problematiche dei piccoli borghi è che spesso non vi sono presenti negozi di alimentari. In questo senso, i soci della cooperativa si preoccupano non solo di richiedere a chi non ha modo di spostarsi dal borgo se c'è necessità di qualcosa, ma anche poi di consegnarglielo non appena possibile. A volte questa cosa può creare un po' di riluttanza da parte degli anziani, che possono sentirsi a disagio nel chiedere tale aiuto. In questo contesto, lo scopo di questo progetto è la creazione di un software per la gestione della spesa alimentare (e non) nel borgo a partire da un set predefinito e ampliabile di prodotti di prima necessità (pane, latte, uova, farina). Gli anziani del paese potranno richiedere ciò di cui hanno bisogno nei seguenti modi:

IN CLASSE

asd

Progetto 3: Viaggi cooperativi

Descrizione: I borghi sono posti meravigliosi. Una delle loro caratteristiche è la lontananza dalle città. Questa loro affascinante peculiarità si scontra però con la praticità nello svolgere anche le più semplici attività. Inoltre spesso questi posti sono abitati da persone anziane e non automunite e i servizi pubblici non sono presenti. L'obiettivo di questo progetto è quello di fornire uno strumento software per permettere ad abitanti dello stesso borgo, o di borghi vicini, di mettersi in contatto in modo da ottimizzare i viaggi verso la città. In questo senso, chi deve andare in città ed è automunito, pubblicherà sull'applicazione il suo evento con orario di partenza e di ritorno, nonché la disponibilità a trasportare persone dei borghi vicini. Allo stesso tempo, chi non è munito di mezzi di trasporto potrà inserire un annuncio di richiesta di passaggio indicando la fascia oraria nella quale deve permanere in città. Le richieste che non potranno essere soddisfatte saranno prese in carico dai soci della cooperativa o da volontari. Anche in questo caso gli anziani potranno usufruire del servizio in due modi differenti:

1. Recandosi in uno dei punti di riferimento/aggregazione del borgo comunicando le proprie necessità all'operatore preposto.
2. Chiamando uno dei soci della cooperativa o volontario il quale i) pubblica l'annuncio sull'applicazione ii) prenota la condivisione dell'auto al posto dell'anziano iii) una volta effettuata la prenotazione, avverte l'anziano.

IN CLASSE

asd

Availability and reliability

Reliability

- The probability of **failure-free system operation over a specified time** in a given environment for a given purpose
- Reliability is concerned with conformance to a given specification and delivery of service

Availability

- The probability that a system, at a point in time, will be **operational and able to deliver the requested services**
- **Both can be expressed in % (e.g., 99% of availability or reliability)**

asd



Reliability metrics

Mean time Between Failure (MTBF)

Mean time to Failure (MTTF)

Mean time to Repair (MTTR)

Availability = $MTBF / (MTBF + MTTR)$

High reliability implies high availability.
The opposite is not always true

asd

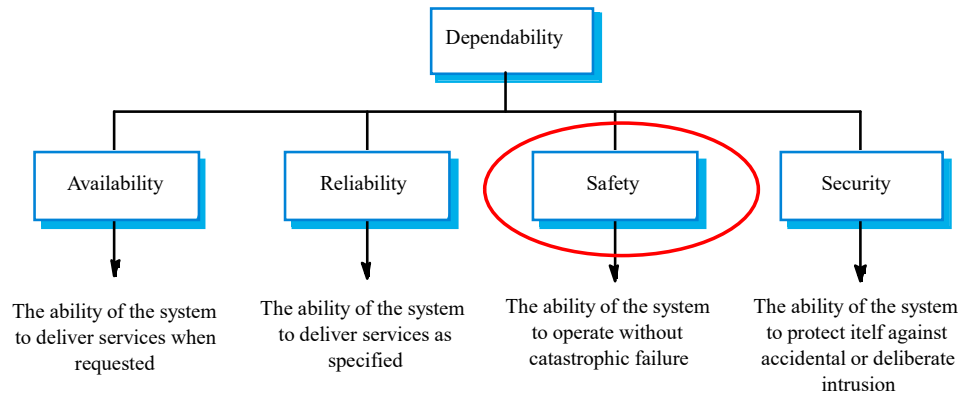
Complexity in Requirements

Sometimes, systems must run reliably for 99.9999 % of the time



asd

Dimensions of dependability



asd

Safety

Safety is a property of a system that reflects the **system's ability to operate**, normally or abnormally, **without danger** of causing human injury or death and **without damage** to the system's environment

It is increasingly important to consider software safety as **more and more devices incorporate software-based control systems**

Safety requirements are **exclusive requirements** i.e. they exclude undesirable situations rather than specify required system services

asd

Safety and reliability

Safety and reliability are related but distinct

→ In general, **reliability and availability are necessary but not sufficient conditions for system safety**

Reliability is concerned with conformance to a given specification and delivery of service

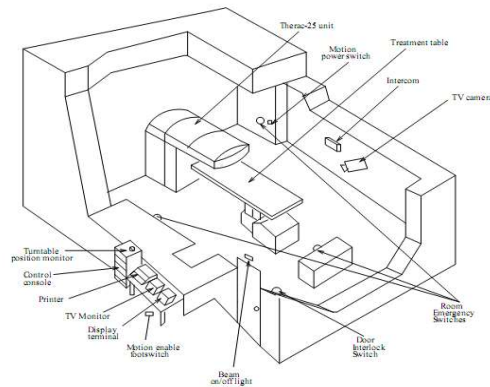
Safety is concerned with **ensuring system cannot cause damage** irrespective of whether or not it conforms to its specification

asd

Safety Failures

Therac-25 safety failure:

- approximately 100 times the intended dose of radiation
- 3 people died. and 6 got injured



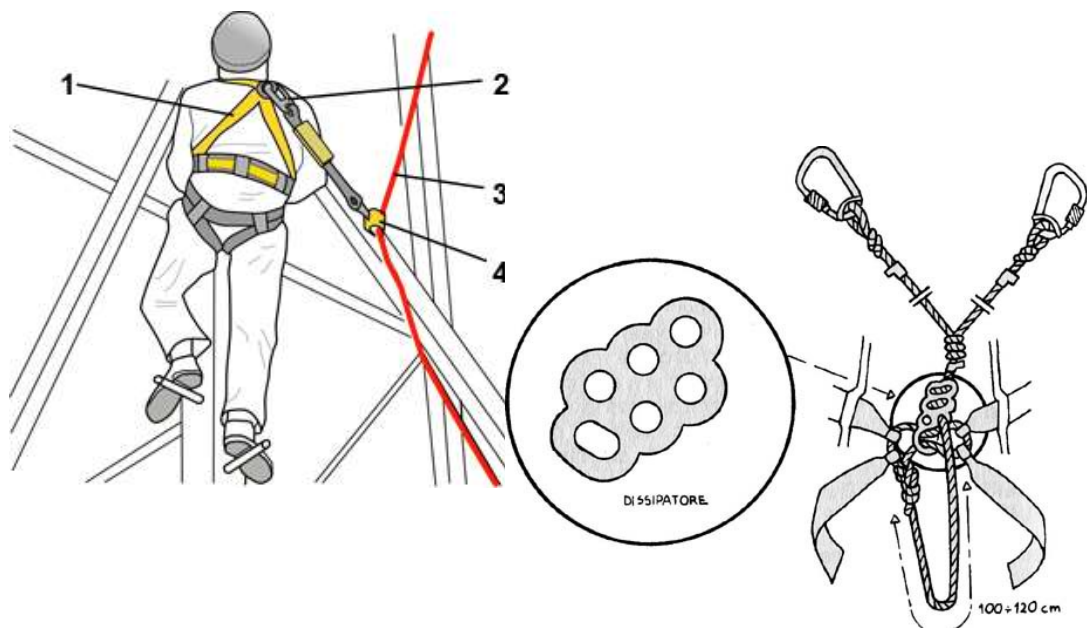
see article at: <http://sunnyday.mit.edu/papers/therac.pdf>

Factors:

- Overconfidence in Software
- Confusing reliability with safety
- Lack of defensive Design
- Failure to eliminate fault causes
- Inadequate software engineering practices
- ...

asd

Example



asd

Safety terminology

Term	Definition
Accident	An unplanned event or sequence of events which results in human death or injury, damage to property or to the environment. A computer-controlled machine injuring its operator is an example of an accident.
Hazard	A condition with the potential for causing or contributing to an accident. A failure of the sensor that detects an obstacle in front of a machine is an example of a hazard.
Damage	A measure of the loss resulting from an accident. Damage can range from many people killed as a result of an accident to minor injury or property damage.
Hazard severity	An assessment of the worst possible damage that could result from a particular hazard. Hazard severity can range from catastrophic where many people are killed to minor where only minor damage results.
Hazard probability	The probability of the events occurring which create a hazard. Probability values tend to be arbitrary but range from <i>probable</i> (say 1/100 chance of a hazard occurring) to implausible (no conceivable situations are likely where the hazard could occur).
Risk	This is a measure of the probability that the system will cause an accident. The risk is assessed by considering the hazard probability, the hazard severity and the probability that a hazard will result in an accident.

asd

Dashboard di Monitoraggio e Gestione delle Alluvioni

Obiettivo:

Realizzare una dashboard che permetta di monitorare in tempo reale dati provenienti da sensori distribuiti in zone a rischio alluvione all'interno di una città o di una regione. Questi sensori rileveranno parametri chiave come livello dell'acqua, intensità della pioggia, saturazione del suolo e velocità del vento. Ogni sensore avrà un codice identificativo univoco e sarà associato a una zona specifica soggetta a rischio idrogeologico.

Descrizione del sistema:

I sensori, distribuiti in diverse aree geografiche come fiumi, canali, bacini di raccolta e aree urbane a rischio, raccoglieranno periodicamente i seguenti parametri:

- **Livello dell'acqua**
- **Velocità di flusso dei fiumi**
- **Pioggia cumulativa**
- **Saturazione del terreno**
- **Velocità e direzione del vento**

Funzionalità base:

- Visualizzazione dei dati
- Possibilità di selezionare un'area geografica specifica e monitorare i sensori associati a tale area.
- Notifica di malfunzionamento dei sensori
- Invio di allarmi prioritari in caso di pericolo imminente, come il rischio di esondazione in corso.

Ogni sensore invierà periodicamente i dati al sistema centrale insieme allo stato di funzionamento del sensore stesso (0-1). La dashboard mostrerà i dati raccolti e informerà i gestori di eventuali superamenti delle soglie critiche predefinite, come ad esempio il superamento del livello di allerta di un fiume o l'intensità delle piogge.

IN CLASSE

asd



Progetto 2: *La tua spesa, il nostro click!* (1/2)

Ambito d'azione: welfare

Contesto: singolo borgo (per iniziare)

Descrizione: Per chi vive in un borgo, anche le attività più semplici possono risultare complicate. L'obiettivo di questo progetto è aiutare le persone più anziane residenti a sviluppare autonomia e fiducia nei confronti dei soci della cooperativa. Da quanto ci è stato raccontato, una delle problematiche dei piccoli borghi è che spesso non vi sono presenti negozi di alimentari. In questo senso, i soci della cooperativa si preoccupano non solo di richiedere a chi non ha modo di spostarsi dal borgo se c'è necessità di qualcosa, ma anche poi di consegnarglielo non appena possibile. A volte questa cosa può creare un po' di riluttanza da parte degli anziani, che possono sentirsi a disagio nel chiedere tale aiuto. In questo contesto, lo scopo di questo progetto è la creazione di un software per la gestione della spesa alimentare (e non) nel borgo a partire da un set predefinito e ampliabile di prodotti di prima necessità (pane, latte, uova, farina). Gli anziani del paese potranno richiedere ciò di cui hanno bisogno nei seguenti modi:

IN CLASSE

asd

Progetto 3: Viaggi cooperativi

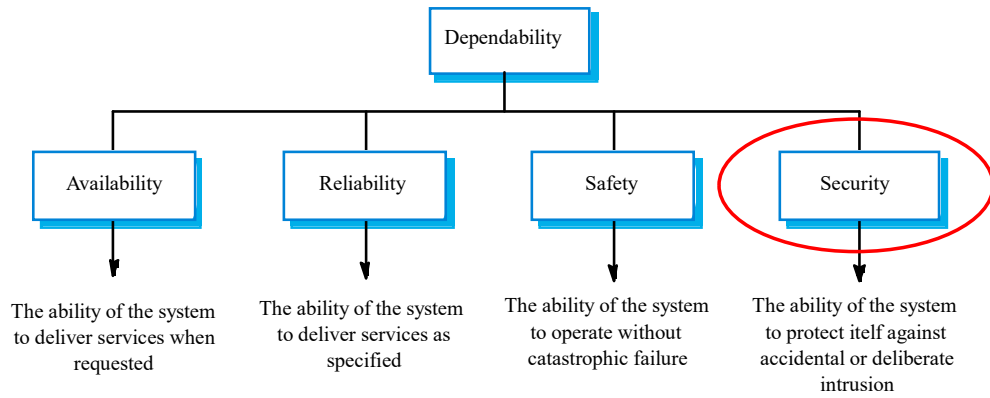
Descrizione: I borghi sono posti meravigliosi. Una delle loro caratteristiche è la lontananza dalle città. Questa loro affascinante peculiarità si scontra però con la praticità nello svolgere anche le più semplici attività. Inoltre spesso questi posti sono abitati da persone anziane e non automunite e i servizi pubblici non sono presenti. L'obiettivo di questo progetto è quello di fornire uno strumento software per permettere ad abitanti dello stesso borgo, o di borghi vicini, di mettersi in contatto in modo da ottimizzare i viaggi verso la città. In questo senso, chi deve andare in città ed è automunito, pubblicherà sull'applicazione il suo evento con orario di partenza e di ritorno, nonché la disponibilità a trasportare persone dei borghi vicini. Allo stesso tempo, chi non è munito di mezzi di trasporto potrà inserire un annuncio di richiesta di passaggio indicando la fascia oraria nella quale deve permanere in città. Le richieste che non potranno essere soddisfatte saranno prese in carico dai soci della cooperativa o da volontari. Anche in questo caso gli anziani potranno usufruire del servizio in due modi differenti:

1. Recandosi in uno dei punti di riferimento/aggregazione del borgo comunicando le proprie necessità all'operatore preposto.
2. Chiamando uno dei soci della cooperativa o volontario il quale i) pubblica l'annuncio sull'applicazione ii) prenota la condivisione dell'auto al posto dell'anziano iii) una volta effettuata la prenotazione, avverte l'anziano.

IN CLASSE

asd

Dimensions of dependability



asd

Security

The security of a system is a system property that reflects **the system's ability to protect itself from accidental or deliberate external attack**

Security is becoming increasingly important as **systems are networked** so that external access to the system through the Internet is possible

asd



MORE ON DEPENDABILITY NFR



asd

CRITICAL SYSTEMS

Safety-critical systems

- Failure results in loss of life, injury or damage to the environment;
- Chemical plant protection system.

Mission-critical systems

- Failure results in failure of some goal-directed activity;
- Spacecraft navigation system.

Business-critical systems

- Failure results in high economic losses;
- Customer accounting system in a bank.

asd

Other dependability properties

Repairability

- Reflects the extent to which the system can be repaired in the event of a failure

Maintainability

- Reflects the extent to which the system can be adapted to new requirements

Survivability

- Reflects the extent to which the system can deliver services whilst under hostile attack

Fault tolerance

- Reflects the extent to which faults can be avoided and tolerated

asd

Fault Tolerance

Definitions:

- Software Fault Tolerance is concerned with techniques designed to tolerate the effects of faults [Anderson&Lee_Book]
- Fault tolerance means that the system can continue in operation in spite of software failure [Sommerville]
- Fault-tolerance or graceful degradation is the property of a system that continues operating properly in the event of failure of some of its parts [Wikipedia]

asd

Fault tolerance failures

Trains in the Netherlands (march 22, 2012)

Tens of thousands of people around the large cities weren't able to travel by train Thursday morning. No trains from and to Amsterdam and Airport Schiphol from early morning until after the morning rush hour. A failure in the back-up system was the cause. ProRail said that there was a fault in the 'switch software'. The system therefore didn't start. And then the signals and switches could not be operated.

Checking some other articles, it simply tries to say that both primary and backup failed, hence no operations.

Links:

<http://www.elsevier.nl/web/Nieuws/Nederland/334086/Oorzaak-van-treinstoring-blijkt-fout-in-software.htm>

<http://www.rnw.nl/english/bulletin/trains-amsterdam-running-again>

On impact on people:

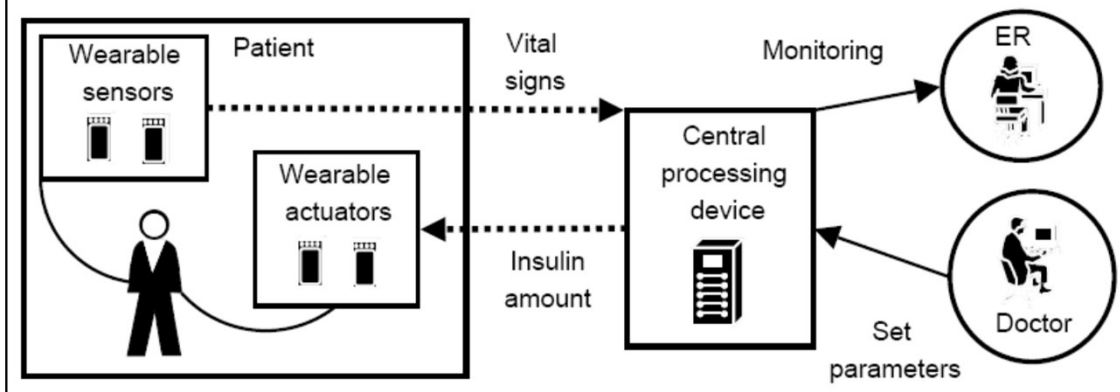
http://www.dutchnews.nl/news/archives/2012/03/signalling_problems_cause_rail.php

asd

Example [REFT05]

The Diabetes Control System

- **1. Sensor stops (E1 or E2):** a wearable sensor could not send valid values (1) try again getting the value to continue the cycle, but if the problem persists (2) stops the delivery and ring the danger alarm.
- **5. Delivery stops (E7, E8):** the sensor that is monitoring an actuator has detected that it did not inject (1) try again to deliver the insulin, but if the problem goes on, (2) stops the delivery and ring the danger alarm.



asd

Dependability vs performance



High levels of dependability can only be achieved at the expenses of system performance

Dependable software includes **extra**, often **redundant**, code to perform the necessary **checking** for exceptional system states and to recover from system faults

asd

Dependability vs cost



Dependability costs tend to increase **exponentially** as increasing levels of **dependability** are required

There are two reasons for this

- The use of **more expensive development techniques and hardware** that are required to achieve the higher levels of dependability
- The **increased testing and system validation** that is required to provide to the client the required levels of dependability (also to prove that they have been achieved)

asd

Dependability vs cost

Because of very high costs of dependability achievement, it may be more **cost effective** to **accept untrustworthy systems** and pay for **failure costs**

However, this depends on social and political factors. A **reputation** for products that can't be trusted **may lose future business**

Depends on **system type** - for business systems in particular, modest levels of dependability may be adequate

asd

Domande

Cos'è la dependability

Standard ISO 25000

NFR 1 vs NFR 2

Critical Systems

Relazioni tra requisiti non funzionali

Scelta dei NFR più rilevanti

...

asd

www.menti.com



Mentimeter

Please enter the code

Submit

The code is found on the screen in front of you

asd