

DOMANDE DEI TEST FATTI IN AULA (con correzione prof)

TEST 1

1. Che cos'è un prodotto software?

- **un'applicazione (o insieme di applicazioni) sviluppata per molti clienti**
- uno strumento di produzione di applicazioni software
- un insieme di ruoli e attività che regolano lo sviluppo di un sistema software
- un programma informatico che risponde ai bisogni dei suoi utenti

2. Che cos'è il ciclo di vita del software?

- la durata di un processo di sviluppo software
- il periodo di tempo che un software impiega ad essere costruito
- **l'insieme delle attività di produzione e gestione di un sistema software**
- la documentazione di un prodotto software

3. Quale tra queste è una proprietà esterna di un sistema software?

- la lunghezza del codice che implementa il sistema
- l'architettura del sistema
- il ciclo di vita del sistema
- **la garanzia che il venditore dà al compratore**

4. Quale tra queste è una proprietà interna di un sistema software?

- il prezzo di vendita del sistema
- il tipo di piattaforma su cui gira il software
- **il periodo di tempo necessario a costruire il sistema**
- il tipo dei dispositivi di input collegabili al sistema

5. Il mio iPhone mi chiede di aggiornare un'app per migliorare la stabilità:

- manutenzione correttiva
- manutenzione adattiva
- **manutenzione preventiva**
- manutenzione correttiva

6. Che cos'è la certificazione del software?

- **un'attività di validazione rispetto ai requisiti di legge**
- un'attività di testing rispetto ai requisiti di legge
- un'attività di verifica rispetto ai requisiti di legge
- un'attività di autenticazione rispetto ai requisiti di legge

7. Qual'è la fase più costosa del ciclo di vita?

- progettazione
- codifica
- deployment
- manutenzione

8. Qual'è la fase più costosa dello sviluppo?

- progettazione
- codifica
- **testing**
- deployment

9. Se diciamo che un prodotto software è di buona qualità, intendiamo che:

- **il prodotto software soddisfa la sua specifica**
- il prodotto software è esente da difetti

- il processo di sviluppo del prodotto ha un buon grado di maturità
- il codice sorgente non contiene errori

10. La manutenzione "perfettiva" del software

- si effettua in seguito a modifiche dell'ambiente in cui opera il software
- corregge errori trovati dopo la consegna del prodotto al committente
- aggiunge funzioni non previste dai requisiti iniziali
- modifica il sistema software prima che gli utenti rilevino un errore

11. Quali sono le fasi tipiche del ciclo di vita secondo lo standard IEEE 12207?

- cascata, iterazione, correzione, valutazione, operazione
- **specificazione, progettazione, codifica, testing di integrazione, testing di sistema**
- elicitazione, validazione, disegno, debugging, monitoring
- acquisition, supply, development, operation, maintenance

12. Quale tipo di manutenzione deriva dalla ridefinizione dei requisiti cliente?

- **perfettiva**
- adattiva
- correttiva
- preventiva

13. In quale fase del ciclo di vita gli errori sono più pericolosi e costosi da commettere?

- **analisi dei requisiti**
- progettazione
- codifica
- manutenzione

14. Quanto vale all'incirca un mese-persona?

- circa 40 ore
- **circa 150 ore**
- circa 300 ore
- circa 720 ore

15. Che cos'è un anno-persona?

- unità di misura del tempo di sviluppo
- unità di misura della durata dello sviluppo
- **unità di misura dello sforzo di sviluppo**
- unità di misura del processo di sviluppo

TEST 2

1. Che cos'è un processo software?

- un'applicazione (o insieme di applicazioni) sviluppata per molti clienti
- uno strumento di produzione di applicazioni software
- **un insieme di attività e ruoli che costruiscono un sistema software**
- un programma che risponde ai bisogni degli utenti

2. Che differenza c'è tra ciclo di vita del software e processo software?

- **il ciclo di vita include il processo software**
- il processo software include il ciclo di vita
- il ciclo di vita è iterativo, il processo lineare
- il ciclo di vita si occupa della documentazione, il processo del codice

3. Quale tra queste è una proprietà importante di ogni modello di processo?

- la durata (in mesi)
- lo sforzo (in mesi-persona)
- **la precisione nella descrizione di ruoli e attività**
- la pianificabilità del processo, cioè l'anticipazione dei risultati conseguibili

4. Qual è la differenza principale tra processi iterativi e processi agili?

- i processi iterativi sono ciclici, i processi agili sono lineari
- **i processi iterativi sono pianificati, i processi agili sono guidati dai test**
- i processi iterativi diminuiscono i rischi, i processi agili li aumentano
- i processi iterativi privilegiano il software, i processi agili la documentazione

5. Quale tra queste NON è un modello di processo software di tipo lineare?

- V Model
- Sawtooth
- Sharktooth
- **RUP**

6. Qual'è la differenza più importante tra i modelli di processo lineari e iterativi?

- i processi basati su modelli iterativi sono più rischiosi
- i processi basati su modelli lineari sono più costosi
- **i processi basati su modelli iterativi sono più lunghi**
- i processi basati su modelli lineari sono più agili

7. Quale tra questi è un principio etico del movimento agile?

- specificare bene i requisiti
- **reagire al cambiamento**
- seguire un piano di processo
- creare la documentazione migliore possibile

8. Quale tra queste pratiche è tipica di Extreme Programming?

- programmazione continua
- rilasci continui
- **integrazione continua**
- rifattorizzazione continua

9. Quale tra queste discipline NON appartiene a RUP?

- business modelling
- design
- configuration management
- **quality assurance**

10. Quale tra queste frasi NON si applica agli artefatti RUP

- **ogni artefatto è il risultato di una singola specifica attività**
- ogni artefatto è responsabilità di uno specifico ruolo
- ogni artefatto è soggetto al controllo delle versioni
- ogni modello è un artefatto

11. Nel RUP, quale tra queste è una disciplina di supporto?

- business modelling
- analysis and design
- **environment**
- requirements

12. Quali tra questi documenti si sviluppa durante la transition?

- Modello di dominio
- **beta test**
- modello del sistema
- piano di progetto

13. Quale tra queste affermazioni è falsa?

- il RUP è un processo di sviluppo in cui si usano diagrammi
- il RUP è un modello di processo di sviluppo di tipo iterativo
- il RUP è un modello di processo di sviluppo proprietario
- **il RUP include una fase che si chiama "verifica"**

14. In quale fase del RUP tipicamente si procede al deployment?

- Inception
- Elaboration
- Construction
- **Transition**

15. Quale tra questi è un indicatore di processo?

- LOG
- numero di difetti per KLOG
- .. di una relazione
- **LOG prodotti in un mese**

TEST 3

1. Quale di questi requisiti è scritto meglio?

- il videogame si può giocare in 2
- due giocatori alla volta possono giocare
- **il videogame dovrà far giocare due giocatori**
- questo videogame si fa in coppia

2. Quale tra questi requisiti è non-funzionale?

- il gioco dovrà supportare uno o due giocatori
- il gioco dovrà permettere ad ogni giocatore di scegliere la squadra
- **il gioco dovrà girare sia su Windows sia su MacOS**
- il gioco dovrà permettere di osservare il computer giocare contro se stesso

3. In un diagramma di casi d'uso cosa rappresenta un attore?

- un'interfaccia utente
- una persona che può usare il sistema
- **un ruolo che definisce un insieme di possibili interazioni**
- un altro sistema che può interagire con quello che stiamo descrivendo

4. Quale tra questi è un requisito funzionale?

- il sistema dovrà funzionare senza errori o con max un errore al mese
- il sistema dovrà funzionare su tutte le principali piattaforme
- il sistema dovrà funzionare tra i 5 e 50 gradi centigradi
- **il sistema dovrà entrare in funzione quando la temperatura tocca i 5 gradi**

5. Quali tra questi requisiti rappresenteresti con un caso d'uso?

- quando un lettore chiede un libro gli viene mostrato il codice di scaffale
- **un lettore può chiedere un libro se è iscritto alla biblioteca**

- un lettore può chiedere un libro se ha restituito quello che aveva
- quando un lettore chiede un libro gli viene detto se c'è

6. Cos'è una feature?

- la descrizione di un requisito come visto da uno stakeholder
- una proprietà di un sistema desiderata da un utente
- la parte interessata ad un requisito
- **una caratteristica di un sistema**

7. Quali tra questi requisiti si mappano direttamente sul codice di un sistema?

- **requisiti architetturali**
- requisiti funzionali
- requisiti non funzionali
- vincoli implementativi

8. In uno scenario di un caso d'uso quale tra questi elementi NON troveremo?

- Attore primario
- attore secondario
- sistema
- **interfaccia utente**

9. In quale tra queste situazioni usereste A <<extends>> B?

- A è un requisito che include B come caso particolare
- B è un requisito che include A come caso particolare
- A è un requisito che usa B
- **A è un requisito opzionale aggiuntivo a B**

10. In quale tra queste situazioni usereste A <<includes>> B?

- A è un requisito che include B come caso particolare
- B è un requisito che include A come caso particolare
- **A è un requisito che usa B**
- B è un requisito opzionale aggiuntivo a A

11. Gli elementi ovali di un "diagramma dei casi d'uso" rappresentano

- **funzioni del sistema**
- oggetti del sistema
- interfacce del sistema
- utenti del sistema

12. Che cos'è l'elicitazione dei requisiti?

- l'analisi dei requisiti
- **la raccolta dei requisiti**
- il tracciamento dei requisiti
- il soddisfacimento dei requisiti

13. Quale tra queste è una tecnica di elicitazione dei requisiti?

- validazione
- tracciamento
- raffinamento
- **prototipazione**

14. Cos'è uno scenario?

- un insieme di requisiti associato ad un bisogno dell'utente
- una feature associata ad un insieme di requisiti

- un caso d'uso associato ad una feature
- **una sequenza di interazioni associata ad un caso d'uso**

15. In un caso d'uso

- gli attori fanno parte del sistema
- gli attori sono oggetti
- **gli attori interagiscono con il sistema**
- gli attori rappresentano esseri umani che usano il sistema

TEST 4

1. Quale tra queste viste non fa parte del modello di un applicazione?

- vista funzionale
- vista strutturale
- vista comportamentale
- **vista dell'interfaccia grafica**

2. Il design di un sistema sw è di buona qualità quando i suoi moduli espongono

- bassa coesione e basso accoppiamento
- bassa coesione e alto accoppiamento
- **alta coesione e basso accoppiamento**
- alta coesione e alto accoppiamento

3. Un modulo è coeso quando

- ha poche dipendenze
- ha tante dipendenze
- offre molte funzioni adatte per diverse situazioni
- **offre poche funzioni, tutte funzionalmente correlate**

4. Un modulo è fortemente accoppiato quando

- ha poche dipendenze da altri moduli
- **ha tante dipendenze da altri moduli**
- non ha dipendenze
- offre molte funzioni per diverse situazioni

5. Quale tra questi è un esempio di modulo?

- un oggetto in java
- un metodo java
- **una classe java**
- un'interfaccia java

6. La relazione A IS_A B (ereditarietà) tra la classe A e la classe B

- **definisce una dipendenza da A a B**
- definisce una dipendenza da B a A
- non è una dipendenza
- definisce una dipendenza doppia da A a B e viceversa

7. La riusabilità di un modulo M

- **è una condizione necessaria per il suo riuso**
- è una condizione sufficiente per il suo riuso
- è una forma di accoppiamento
- è una forma di dipendenza

8. Quale tra questi è il più alto livello di coesione?

- **logico**
- temporale
- procedurale
- comunicativo

9. Quale tra questi è il più alto livello di accoppiamento?

- **accoppiamento sui dati**
- accoppiamento di controllo
- accoppiamento esterno
- accoppiamento sui contenuti

10. Quale tra queste è una tipica relazione gerarchica tra moduli?

- il modulo A è più grande del modulo B
- **il modulo A è contenuto nel modulo B**
- il modulo A è accoppiato col modulo B
- il modulo A è una versione del modulo B

11. Come si misura il grado di accoppiamento tra due moduli A e B?

- contando le dipendenze dirette tra A e B
- contando le dipendenze dirette tra A e quelle indirette con altri moduli
- **analizzando la forma di accoppiamento (es. accoppiamento di contenuto)**
- analizzando quanto sono coesi A e B

12. Cos'è un modello di dominio?

- un diagramma delle classi
- un diagramma degli oggetti
- **un diagramma delle classi più importanti**
- un diagramma degli oggetti più importanti

13. Quale tra queste è una "responsabilità" in una descrizione di un'agenda?

- la struttura dell'agenda in mesi e settimane e giorni
- la collaborazione dell'agenda con il client di posta elettronica
- il tipo "data" con la definizione dei tipi mese, giorno, settimana
- **l'elenco degli appuntamenti in agenda**

14. In una scheda CRC quale tra questi elementi NON troviamo?

- il nome della classe
- le responsabilità della classe
- **gli attributi della classe**
- le collaborazioni della classe

15. A cosa serve il diagramma di sequenza (SSD)?

- descrive i messaggi interni al sistema
- in un caso d'uso, descrive gli eventi generati dagli attori esterni
- descrive i messaggi tra due sistemi
- **descrive i messaggi tra attori ed il sistema**

TEST 5

1. Una classe di cui può esistere nel sistema una sola istanza

- command
- **singleton**
- mediator
- proxy

2. Una sola classe rappresenta l'interfaccia di un intero sottosistema

- singleton
- mediator
- **facade**
- abstract factory

3. Crea un'istanza di parecchie famiglie di classi

- **abstract factory**
- builder
- factory method
- decorator

4. Accede sequenzialmente agli elementi di una collezione

- chain of responsibility
- **iterator**
- decorator
- visitor

5. Notifica un evento o cambiamento di stato ad un insieme di oggetti

- mediator
- prototype
- state
- **observer**

6. Separa l'astrazione di un oggetto dalla sua implementazione

- decorator
- **bridge**
- abstract factory
- state

7. Incapsula un algoritmo in una classe

- adapter
- command
- **strategy**
- mediator

8. Aggiunge dinamicamente responsabilità a degli oggetti

- **decorator**
- bridge
- proxy
- strategy

9. Cattura (portando all'esterno senza violare l'incapsulamento) e ripristina lo stato interno di un oggetto

- **memento**
- state
- proxy

- observer

10. E' un'istanza del tutto inizializzata, pronta per essere copiata o clonata

- **prototype**
- singleton
- proxy
- visitor

11. Separa la costruzione di un oggetto dalla sua rappresentazione

- proxy
- decorator
- interpreter
- **builder**

12. Un oggetto che rappresenta e fornisce accesso ad un altro oggetto

- **proxy**
- decorator
- interpreter
- builder

13. Definisce una struttura ricorsiva di oggetti semplici o composti

- **composite**
- chain of responsibility
- flyweight
- abstract factory

14. Rende interoperabili interfacce differenti

- interpreter
- mediator
- **adapter**
- facade

15. Valuta le frasi di un linguaggio, mappando ciascun simbolo terminale o non terminale in una classe specifica

- **interpreter**
- proxy
- command
- chain of responsibility

TEST 7

1. Qual'è il vantaggio principale della stima basata su function point analysis

- **disincentiva la stima di produttività basata su LOC inutili**
- è una stima completamente oggettiva, soggetta a regole precise
- è una stima poco costosa e immediata
- disincentiva la stima di costo basata su durata esagerata

2. Qual'è lo svantaggio principale della stima basata su LOC?

- **incentiva la stima di produttività basata su LOC inutili**
- è una stima soggettiva, non soggetta a regole precise
- è una stima costosa
- di solito sottostima lo sforzo complessivo

3. In COCOMO quante ore-persona vale un mese-persona?

- 40 ore-persona
- **152 ore-persona**
- 304 ore-persona
- 1000 ore-persona

4. Che cos'è un anno-persona?

- unità di misura del tempo di sviluppo
- unità di misura della durata dello sviluppo
- **unità di misura dello sforzo di sviluppo**
- unità di misura del processo di sviluppo

5. In un grafico GANTT cosa c'è sull'asse verticale (ordinata)?

- il tempo in mesi
- lo sforzo in mesi-persona
- **le attività del progetto ordinate temporalmente dall'alto verso il basso**
- le attività di progetto ordinate per importanza

6. In un grafico PERT cosa rappresenta un arco dal nodo A al nodo B?

- l'attività per passare da A a B e la sua durata
- il fatto che il task A ha più importanza di B
- **il fatto che A va fatto prima di B**
- il fatto che il task A è più critico di B

7. Cos'è un punto funzionale (function point)?

- **un requisito funzionale dal punto di vista dello sviluppatore**
- un requisito funzionale dal punto di vista dell'utente
- le LOC equivalenti ad un requisito funzionale
- un comando o procedura in un linguaggio di programmazione

8. A cosa serve stimare le linee di codice di un sistema software prima di costruirlo?

- a capire i rischi del progetto
- **a stimare lo sforzo**
- a stimare la durata
- a definire la produttività da conseguire

9. In un'azienda, cosa distingue un progetto dalla gestione quotidiana?

- il progetto ha un budget
- **il progetto deve produrre un risultato in tempi predefiniti**
- il progetto modifica l'organizzazione
- il progetto richiede pianificazione

10. Se puoi usare un solo strumento di gestione, cosa scegli?

- **uno strumento per definire WBS (Work Breakdown Structure)**
- uno strumento per definire grafici GANTT
- uno strumento per definire un PERT
- un registro dei rischi

11. Qual'è la caratteristica più importante per un project manager?

- la capacità tecnica, dev'essere il più bravo dei membri del team
- la diplomazia, deve usare tatto coi membri del team
- **l'autorevolezza, deve avere il potere formale di gestire il team**
- la psicologia, deve sapere dire le cose "giuste" a ciascun membro del team

12. Qualità, durata e costo sono importanti in ogni progetto. Tuttavia
- il costo è più importante di durata e qualità
 - hanno sempre pari importanza
 - dipende da ciascun progetto definire la loro importanza relativa
 - **la qualità è importante per il cliente, costo e durata per il team di sviluppo**
13. Quale tipo di diagramma mostra la struttura di un progetto?
- **WBS**
 - GANTT
 - PERT
 - nessuno di questi
14. Un diagramma PERT serve per calcolare
- la durata massima di un progetto
 - **la durata minima di un progetto**
 - lo sforzo complessivo
 - il costo complessivo di un progetto
15. Il metodo COCOMO stima
- la durata, a partire dalle persone impiegate nel progetto
 - **lo sforzo, a partire dalle linee di codice stimate**
 - lo sforzo, a partire dalle linee di codice effettive
 - il costo, a partire dalla difficoltà del progetto

DOMANDE DEI TEST DI VECCHI COMPITI (con soluzioni mie)

1. Uno tra questi è un requisito, gli altri sono feature. Qual è il requisito?
- Selezione automatica delle città di arrivo avendo selezionato quella di partenza
 - Riutilizzo dei dati dei clienti registrati
 - **Se il cliente non inserisce i dati del documento non viene fatto il check-in**
 - Acquisto di molteplici viaggi tramite carrello elettronico
2. Una sola tra questi è una feature, gli altri sono requisiti. Qual'è la feature?
- L'editor deve permettere di editare formule matematiche
 - **L'editor ha un ambiente per la gestione di formule matematiche**
 - L'editor deve permettere di importare tabelle Excel
 - L'editor deve creare automaticamente l'indice analitico
3. Uno tra questi è un requisito, gli altri sono feature. Qual è il requisito?
- **L'applicazione permette di ingrandire ogni articolo del giornale**
 - Se vuole, l'utente può salvare l'edizione corrente nel suo archivio personale
 - L'applicazione permette di pagare l'abbonamento via AppStore
 - L'applicazione ha un archivio delle 15 edizioni precedenti
4. Una sola tra questi è una feature, gli altri sono requisiti. Qual'è la feature?
- È possibile abbandonare l'applicazione
 - L'utente abbandona l'applicazione cliccando sul bottone nero
 - L'utente abbandona l'applicazione se tocca lo schermo con quattro dita
 - **Se l'utente abbandona l'applicazione appare la schermata "home"**

5. In un diagramma di attività quale tra questi elementi di modellazione NON troviamo?

- Partizioni (swimlane)
- Punti di decisione
- Barre di sincronizzazione
- **Guardie su eventi**

6. In un diagramma di stato quale tra questi elementi di modellazione troviamo?

- Partizioni (swimlane)
- Pseudostati
- Barre di sincronizzazione
- **Guardie su eventi**

7. Come si chiama il testing in cui non si usa il sorgente del programma?

- Whitebox
- **Blackbox**
- Review
- Validazione

1. Uno tra questi è un requisito funzionale, gli altri no. Qual è quello funzionale?

- Se una app si blocca il sistema operativo continua a funzionare
- Se una app si blocca il sistema operativo consuma meno energia
- Se una app si blocca il sistema operativo non viene danneggiato
- **Se una app si blocca è possibile telefonare premendo il tasto home**

2. Una sola tra questi è una feature, gli altri sono requisiti. Qual'è la feature?

- Il telefono permette comunque di chiamare i numeri di emergenza
- Il telefono che riceve una telefonata sospende tutte le app in esecuzione
- Se il telefono accede un application store può scaricare una o più app
- Il telefono che scarica una app può comunque ricevere telefonate

3. Uno solo tra questi è un requisito non funzionale. Qual è?

- **L'applicazione "LaRepubblica" ha un archivio che controlla l'utente**
- Se vuole, l'utente può salvare l'edizione di oggi nel suo archivio personale
- L'applicazione permette di pagare l'abbonamento via AppStore
- L'applicazione può archiviare fino a 15 edizioni diverse

5. Quale tra queste è una proprietà importante di ogni modello di processo software:

- La durata
- Lo sforzo
- **La precisione**
- La pianificabilità

8. Quale tra queste fasi NON fa parte del modello waterfall?

- System allocation process
- Installation process
- Operation & support process
- **Transition process**

4. Quali sono le fasi tipiche del ciclo di vita secondo il RUP?

- **Inception, elaboration, construction, transition**
- Business modeling, requirements, design, coding, deployment
- Debugging, testing, verification, validation

- Acquisition, supply, development, operation, maintenance

9. Che cos'è il deployment?

- L'attività che rende un sistema software pronto all'uso
- L'attività che convince un cliente che il software funziona
- **L'attività che precede la manutenzione**
- L'attività che segue dopo il testing

10. Quando capita che un software si guasta, qual è il primo passo del debugging?

- **Scegliere il debugger**
- Riprodurre il guasto
- Rileggere le specifiche
- Rifare il testing

A che serve il diagramma dei casi d'uso di un sistema?

- A definire gli scenari del sistema
- a definire i sistemi che interagiscono col sistema;
- a definire le interfacce utente del sistema;
- **a definire i requisiti funzionali del sistema**

Quale tra questi diagrammi UML è di tipo strutturale?

- Componenti
- comunicazione
- attività
- casi d'uso

Se A è superclasse di B significa che?

- A dipende da B
- **B dipende da A**
- l'ereditarietà tra A e B non include la dipendenza
- B dipende dagli attributi di A ma non dai suoi metodi

In UML, un diagramma di sequenza si può usare per mostrare

- una sequenza di messaggi sincroni o asincroni
- le dipendenze tra i componenti che compongono il sistema
- come si invocano le interfacce dei componenti
- **una sequenza di eventi e le relative azioni di reazione**

come si esprime l'aggregazione tra classi in un diagramma UML?

- Con un diamante nero
- **con un diamante bianco**
- con una freccia a triangolo vuoto
- con una linea tratteggiata

In un diagramma di stato quali tra questi elementi troviamo?

- Classi e relazioni
- **oggetti e timeline**
- decisioni e attività
- transizioni e azioni

I nodi di un diagramma di stato rappresentano

- oggetti
- **stati**
- attività

- transizioni

a cosa serve stimare le linee di codice di un sistema software prima di costruirlo?

- A capire i rischi del progetto
- **a stimare lo sforzo**
- a stimare la durata
- a definire la produttività da conseguire

quale di questi requisiti è scritto meglio?

- Il sistema può avere uno o più utenti
- il sistema può essere usato da molti utenti simultaneamente
- il sistema dovrà supportare almeno 100 utenti simultanei
- **il sistema dovrà supportare fino a 100 utenti simultanei**

in un diagramma di casi d'uso, cosa rappresenta una linea tra un attore e un ovale?

- Uno scenario
- una persona che usa il sistema
- un flusso di dati
- **un'associazione tra attore e requisiti funzionali**

quale tra questi è un principio tipico dello sviluppo agile?

- Modificare spesso i requisiti funzionali
- **semplificare il codice anche se non necessario**
- produrre nuovi test dopo ogni iterazione
- creare il manuale utente nella prima iterazione

un modulo software implementa un insieme di

- requisiti
- scenari
- **interazioni**
- associazioni

un modulo software ha sempre

- **un'interfaccia**
- un'interfaccia utente
- più versioni
- più configurazioni

nei diagrammi di deployment troviamo

- I moduli con le interfacce e le porte
- le attività dei componenti principali
- **I nodi dell'infrastruttura hardware e le loro relazioni**
- la configurazione principale del sistema