

Programma di Tecnologie e Progettazione di Sistemi Informatici e di Telecomunicazioni

MOTIVAZIONI ED OBIETTIVI

L'impostazione di tutto il programma fa riferimento agli obiettivi specifici di apprendimento proposti dal ministero per questa nuova disciplina, introdotta dall'ultima riforma e pubblicata il 30-3-2012 nel supplemento ordinario n. 60 della GAZZETTA UFFICIALE, e che quest'anno giunge per la prima volta al traguardo degli esami di maturità.

Per quanto attiene alle conoscenze i punti centrali sono stati:

- i metodi e le tecnologie per la programmazione di rete,
- i protocolli ed i linguaggi di comunicazione a livello applicativo,
- le tecnologie per la realizzazione di servizi web.

Il fuoco è stato mantenuto sempre sulle problematiche connesse con la comunicazione.

Per quanto riguarda le abilità e le competenze sono stati individuati i seguenti obiettivi:

- realizzazione di applicazioni base per la comunicazione di rete,
- progettazione essenziale dell'architettura di un prodotto/servizio individuandone le componenti tecnologiche,
- sviluppo di programmi client-server che utilizzano protocolli esistenti,
- progettazione di semplici protocolli di comunicazione,
- realizzazione di semplici applicazioni orientate ai servizi.

Come strumento conduttore, per la realizzazione di tutto il lavoro, è stato utilizzato il linguaggio PHP in ambiente Linux.

Sulla base dell'esperienza e dei risultati conseguiti negli anni precedenti nell'insegnamento di materie affini, il programma è stato svolto privilegiando sempre quegli invarianti culturali indispensabili a continuare la formazione permanente della propria professionalità piuttosto che quei temi caratterizzati soltanto da elevati e aggiornati contenuti tecnologici. In questa ottica, data la specificità della materia, si è tentato di far comprendere, senza soluzione di continuità, l'intera sequenza che va dall'idea di progetto, nella mente dell'operatore, all'attivazione dei vari dispositivi/servizi, passando attraverso tutta la serie di stratificazioni logico-tecnologiche adoperate nell'informatica industriale.

Coerentemente con le scelte di fondo e gli obiettivi indicati, poiché lo scopo primario è quello di far acquisire non solo conoscenze ma anche abilità e competenze, è stata adottata una metodologia che ha privilegiato momenti di scoperta a partire da problemi semplici e stimolanti. Di conseguenza, la strategia didattica adottata ha previsto, da un lato, che l'acquisizione di determinate conoscenze venisse sollecitata attraverso tecniche induttive tendenti alla ricerca della soluzione corretta e, dall'altro, che venissero generate situazioni problematiche per risolvere le quali non fosse richiesta una semplice riproduzione di conoscenze bensì una loro ristrutturazione per effetto dell'intuizione e della scoperta.

La classe presenta, nel complesso, un livello medio tra il sufficiente ed il buono. Il rapporto instauratosi tra insegnanti e studenti è stato sempre corretto ed il lavoro scolastico si è svolto, pertanto, in un clima di serena partecipazione. Il giudizio sui risultati ottenuti è complessivamente positivo, alcuni elementi si sono distinti per tenacia, capacità, intelligenza e diligenza, e grazie ad una buona motivazione hanno approfondito le tematiche più interessanti, talvolta, anche in ambito extrascolastico.

ARGOMENTI TRATTATI

Generalità sui sistemi distribuiti. Benefici della distribuzione. Svantaggi legati alla distribuzione. Architetture a livelli. I modelli di comunicazione. Il modello client-server. Livelli e strati. Applicazioni di rete. Scelta dell'architettura per l'applicazione di rete. Servizi offerti dallo strato di trasporto alle applicazioni. Il web server Apache, configurazione ed uso di Apache in ambiente Linux con cenni all'utilizzo di Apache in ambiente Windows.

La sintassi del linguaggio PHP, tipi di dati principali, costanti e variabili, operatori. Le strutture di controllo principali: i costrutti if, sequenze e iterazioni, cicli for. Funzioni definite dall'utente e passaggio parametri. La funzione isset(). Array sequenziali ed associativi. Array predefiniti: \$_GET e \$_POST. Gestione dei contenuti di un FORM.

Principali operazioni connesse all'utilizzo dei file di testo e funzioni correlate: readfile(), file_get_contents(), file(), fopen(), fread(), file_put_contents(), fwrite(), rename(), copy(), unlink(), file_exists(), is_readable(), is_writable(), is_executable(), is_file(), is_dir(), chmod(), chown(). Uso dei file flat per la gestione di semplici banche dati e funzioni di supporto: explode(), implode(). I socket ed i protocolli per la comunicazione di rete. Le porte di comunicazione ed i socket. La connessione tramite socket. I socket e la comunicazione con TCP/UDP.

Le funzioni socket della libreria BSD: socket_accept(), socket_bind(), socket_close(), socket_connect(), socket_create(), socket_getpeername(), socket_last_error(), socket_listen(), socket_read(), socket_recv(), socket_recvfrom(), socket_select(), socket_send(), socket_sendto(), socket_strerror(), socket_write().

Funzioni socket e/o ausiliarie native PHP: fsockopen(), gethostbyname(), stream_copy_to_stream(), stream_socket_accept(), stream_socket_get_name(), stream_socket_server(), error_reporting().

Creazione di un socket, connessione ad un server remoto, invio dei dati, ricezione di una risposta, chiusura di un socket.

Ottenere l'indirizzo IP dal nome del dominio, effettuare il Bind di un socket, mettere un socket server in ascolto, accettare una connessione, utilizzare telnet come client di prova.

Mantenere un server in ascolto continuo (Live), gestione di un server in modo asincrono con il metodo del polling.

Creazione di un socket UDP, attivazione di un semplice server UDP, attivazione di un semplice client UDP.

Modularità nella struttura di uno script, utilizzo della funzione fsockopen(), uso delle funzioni interne del PHP per i socket, scrittura di un semplice server con le funzioni interne.

Utilizzo del browser come client, sviluppo di uno script per la scansione dei principali servizi di un server, emulazione con uno script del comando ping.

LABORATORIO

Durante le ore di laboratorio, con l'ausilio indispensabile delle strutture in dotazione all'area, la classe, suddivisa in gruppi, ha verificato tutti gli argomenti svolti durante le lezioni teoriche. In particolare, nella seconda parte dell'anno, ciascun gruppo, individuati dei temi ritenuti di particolare interesse sia per le applicazioni sia per i contenuti teorici, ha cercato di approfondirne la conoscenza sviluppando autonomamente delle varianti sui temi proposti.

SUSSIDI DIDATTICI ED ATTREZZATURE

- Sito web dell'insegnante: <http://www.salvorosta.it>
- Libro di testo: Tecnologie e Progettazione di Sistemi Informatici e di Telecomunicazioni, vol. 3. Paolo Camagni e Riccardo Nikolassy, Ediz. Hoepli (ISBN: 978-88-203-5158-8).
- Fotocopie ed appunti, manuali informatici relativi al software presente in laboratorio.
- Laboratorio di Sistemi (aula 12 della scuola) attrezzato con PC Intel i7 multimediali, con sistema operativo Linux a 64 bit, in doppia configurazione client/server, collegati in rete locale e ad Internet.

SPAZI UTILIZZATI

- Aule di didattica
- Laboratorio di Sistemi

TEMPI

132 ore circa.

CRITERI DI VALUTAZIONE

Considerando il livello di partenza dello studente la valutazione di tipo verticistico ha tenuto conto degli standard previsti dai programmi ministeriali. Per le prove orali:

Conoscenza dei contenuti richiesti

Comprensione dei contenuti richiesti

Capacità di applicazioni pratiche dei contenuti richiesti

Linguaggio tecnico

Capacità di utilizzare collegamenti interdisciplinari tra argomenti per le prove pratiche

Capacità di sintetizzare il problema proposto.

STRUMENTI DI VALUTAZIONE

Verifiche orali

Verifiche delle prove pratiche di laboratorio

Accertamento dei processi di maturazione intervenuti in ogni alunno

ATTIVITA' DI POTENZIAMENTO E RECUPERO

Ripetizione e approfondimento degli argomenti svolti per dare modo a tutti gli alunni di assimilarli, e lavoro personalizzato in alcuni casi particolari.