

Tecnologie Web: Introduzione al corso

Prof. Raffaele Montella, Ph.D.

raffaele.montella@uniparthenope.it

Il corso

Nome: Tecnologie Web - TechWeb - TW

Docente: Raffaele Montella

Orario di ricevimento: Stanza 421, Lunedì dalle 11 alle 16, prendere appuntamento su teams.

Crediti formativi: 6

Settore Scientifico Disciplinare: INF/01

Articolazione didattica:

- lezioni: 48h di lezione frontale

Prerequisiti:

E' necessaria la conoscenza degli argomenti di Programmazione, Basi di Dati, Architettura e Reti (ovvero l'esame è fortemente **sconsigliato** al II anno, **consigliato** al III).

Finalità

- Il corso si propone di fornire le conoscenze e le competenze necessarie per la realizzazione di applicazioni web-based, contestualizzandole nell'ambito più vasto delle **applicazioni distribuite**.
- Il corso è un'introduzione alla programmazione di applicazioni in ambiente internet con riferimenti anche alla fruizione attraverso dispositivi mobili e all'**internet delle cose** (Internet of Things).

Particolare attenzione viene posta alla definizione dei modelli su cui questa tipologia di applicazioni si basa, partendo da quelli più tradizionali per arrivare alle evoluzioni recenti.

Piattaforma di riferimento: LAMP (No Windows. No XAMP.)

- Linux
- Apache/Nginx
- MongoDB
- Python

Tecnologie abilitanti:

- Web service
- Cloud computing
- Sensori, attuatori, microcontrollori e microprocessori

Parte significativa del corso è dedicata allo svolgimento di esercitazioni guidate in laboratorio e ad esercitazioni da compiere a casa.

Seguo o non seguo?

Corsisti:

- È considerato corsista chi partecipa a:
 - 2 prove intercorso (e-learning - in presenza)
 - 1 prova al momento dell'esame (e-learning - in presenza)
 - Base voto teoria la media dei 2 voti migliori delle 3 prove.
 - Proposal progetto (5% del voto finale)
 - Progetto (35% demo del voto finale)



Esame entro **ottobre** 2023
Presentazione progetto (40%)
60% Esame teorico.



fallback

Non corsisti:

- Progetto (40% voto finale)
- 1 prova al momento dell'esame (e-learning/proctoring)
- Possono non fare il progetto (nel caso il voto finale dipende solo dall'esame teorico: 60%).



Esame da **gennaio** 2024
[Presentazione progetto (40%)]
60% Esame teorico.

I fuori corso possono scegliere se seguire il corso ed essere considerati corsisti o fare l'esame come non corsista.

Organizzazione

Durata: Settembre – Dicembre 2022

Lezioni frontali

8 novembre: Proposta progetti

14 novembre: Prova intercorso #1

19 dicembre: Prova intercorso #2

23 dicembre: Presentazione progetti

Organizzazione

Orario:

- Lunedì 16:00 – 18.00
- Martedì 16:00 – 18.00

Esercitazioni speciali:

SmartLab - CDN434

- I progetti devono essere consegnati condividendo la url <https://github.com> 2 giorni prima della data di appello.
- I progetti devono essere riproducibili e dimostrabili seguendo le istruzioni contenute nel file README.md
- Tutta la documentazione interna ed esterna deve essere in lingua inglese
- Tutto il codice DEVE essere rilasciato su <https://github.com> con licenza Apache 2.0 / MIT / LGPL
- Date di esame già presenti in piattaforma.

Programma

- Introduzione alle tecnologie alla base del web
- Il linguaggio Python
- Applicazioni web in Python: Flask
- Database No SQL
- Introduzione all'internet delle cose
- Controllo di sensori e attuatori con Arduino
- Applicazioni distribuite su cloud

Il programma dettagliato è nel file Syllabus disponibile su [ugov](#).

Proposal progetti: 8 Novembre, ore 16:00

- È il primo “evento” ufficiale del corso.
- Gruppi composti da 1, 2, o 3 studenti.
Progetti impegnativi possono essere divisi in sottoprogetti.
- Gli studenti devono registrarsi su <https://github.com> con la propria email istituzionale @studenti.uniparthenope.it (se non hanno già un account github studenti).
- Ogni gruppo deve creare un repository progetto (può essere a nome di uno dei componenti del gruppo).
- Il proposal deve essere realizzato con una presentazione o un video di massimo 90 secondi.
- Il proposal può essere presentato da uno solo dei componenti del gruppo.
- Il materiale del proposal deve essere aggiunto al repository del progetto.

Prima prova intercorso: 14 Novembre, ore 16:00

- Le tecnologie web - Il paradigma client-server - Iper testo, multimedia, ipermedia - Il web come documento ipermediale - Il protocollo HTTP, richieste, risposte, intestazioni, cookie e mantenimento delle sessioni - Server web.
- Caratteristiche generali di una pagina web -Elementi ed attributi - Struttura base del documento -Testo, iper testo, immagini, liste ed elenchi, form, tabelle, collegamenti e scripting lato client - Il cascading style sheets (CSS): introduzione.

Prima prova intercorso: 14 Novembre, ore 16:00

- I client per il web, browser, altri client, le applicazioni mobile.
- Programmazione server side e client side
- Linguaggi di descrizione per le pagine web - Extensible Markup Language - Il markup - XML come metalinguaggio - Caratteristiche sintattiche - Documenti XML ben formati - HTML 5

Prima prova intercorso: 14 Novembre, ore 16:00

- Javascript, elementi di base ed avanzato.
- Il Document Object Model, uso ed applicazioni.
- Cascade Style Sheet.

Seconda prova intercorso: 20 Dicembre, ore 16:00

- Le Progressive Web App
- Ajax e jQuery
- Introduzione al linguaggio Python - costrutti, variabili, tipi di dati - Python argomenti avanzati: classi, oggetti e namespace - Python come linguaggio per le tecnologie web: introduzione a Flask - Generare pagine web con Flask.

Seconda prova intercorso: 20 Dicembre, ore 16:00

- I web service - SOAP e REST - JSON: Javascript Object Notation - GET, POST e gli altri metodi HTTP in Flask.
- Database Non Relazionali (MongoDb)
- Python & Flask per lo sviluppo di API REST

Materiale didattico

- Lucidi delle lezioni: <https://elearning.uniparthenope.it/course/view.php?id=121>
- Google Drive:
<https://drive.google.com/drive/folders/1ZzuSYq9QWAYYYXiQhvsGDSaCEZqJRC7gE?usp=sharing>
- GitHub: <https://github.com/informatica-uniparthenope/TW6>
- Libri di testo:
 - Programmazione Web Lato Server - ultima versione aggiornata - V. Della Mea, L. Di Gaspero, I. Scagnetto. Apogeo
 - Flask Web Development: Developing Web Applications with Python - Miguel Grinberg
 - Learning Python - Mark Lutz
 - Cloud computing per applicazioni web - J. Barr. Apogeo
 - Arduino: Guida completa - Paolo Aliverti

Altre risorse

- Portale W3Schools: <https://w3schools.com> (in inglese)
- Portale HTML.it: <https://www.html.it> (in italiano)
- Flask Mega Tutorial: <https://blog.miguelgrinberg.com/post/the-flask-mega-tutorial-part-i-hello-world> (in inglese)
- Flask API con JWT authentication: <https://aaronluna.dev/series/flask-api-tutorial/overview/> (in inglese)

Valutazione

- Il progetto è proposto dallo studente o dal gruppo di studenti in numero n con $1 \leq n \leq 3$.
- Una presentazione di massimo 3 slide e/o 90 secondi (Elevator Pitch) deve convincere che il progetto è originale (no riciclo) ed innovativo (deve fare una cosa nuova o una vecchia in modo nuovo e migliore).
- Il docente emenda, rigetta o approva il progetto.

Valutazione

- Al momento della dimostrazione del progetto, questo deve essere accessibile attraverso la rete Internet. È possibile far funzionare il progetto su proprio hardware, se questo è raggiungibile dall'esterno.
- Tutto il materiale deve essere pubblico su github.com già durante lo sviluppo del progetto.
- Tutto quanto sviluppato deve essere pubblicato come open source con licenza preferibilmente Apache 2.0.

Valutazione

La valutazione è effettuata con i seguenti pesi: progetto 40%, orale 60%

- Corsisti: 15%: Presentazione del progetto, massimo $3+3*n$ slide (Valutazione singola - un minuto per slide)
- Non corsisti: 20%: Presentazione del progetto, massimo $3+3*n$ slide (Valutazione singola - un minuto per slide)

20%: Dimostrazione del progetto; massimo 5 minuti (Valutazione di gruppo)

60%: Verifica delle conoscenze teorico/pratiche (Valutazione singola)

Progetti

- Tutti i progetti devono rispettare le linee guida relative all'etica dell'Università degli Studi di Napoli "Parthenope"
- I progetti possono anche coprire più esami (va concordato con i docenti) in modo da avere un unico obiettivo per semestre.
- I migliori progetti saranno considerati per la pubblicazione come Web App/API ufficiali di Ateneo.

Primo contatto

- Chi ha esperienze di Tecnologie Web?
- Quanti linguaggi di programmazione conoscete?
- Chi ha già avuto esperienze con Python?
- Cos'è il cloud computing?
- Che percezione avete dell'Internet delle Cose?
- Cosa sono i web service?
- Chi ha già “giocato” con Arduino?
- Cos'è un database?
- Chi compra su Amazon?

Hobby? Passioni? Passatempi (apparte lo studio!)

Ambiente di sviluppo e altre risorse

- Personal computer: Linux, Ubuntu 20.04 LTS
- Server: Linux, Ubuntu 20.04 LTS
- Amazon Web Services - <https://aws.amazon.com/it/free/>
- Risorse UNP
- Github
- ArduinoIDE

Software di virtualizzazione: VirtualBox

Gestore di container: Docker

```
docker run \
  --name ubuntu \
  -e HOST_IP=$(ifconfig en0 | awk '/ *inet /{print $2}') \
  -v /Users/hudson/Workspaces:/src \
  -t -i \
  ubuntu /bin/bash
```

Ubuntu con Docker su MacOS

Tutorial per Windoes & MacOS

<https://medium.com/@hudsonmendes/docker-have-a-ubuntu-development-machine-within-seconds-from-windows-or-mac-fd2f30a338e4>

Creative Commons & Google Drive

Queste slide e tutto il materiale didattico sarà sviluppato in una cartella Google Drive condivisa con gli studenti che potranno apportare il loro contributo.

Periodicamente il materiale sarà finalizzato e pubblicato sul portale e-learning di Dipartimento.

Gli studenti possono essere abilitati all'accesso della cartella Google Drive inviando la propria gmail (o altra casella di posta associata al servizio) a:

raffaele.montella@uniparthenope.it

Tutte le comunicazioni con il docente continuano ad avvenire attraverso l'email istituzionale @studenti.uniparthenope.it

Domande e risposte

...