



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# **Il progetto del corso di Tecnologie Web - A.A. 2021/22**

**Angelo Di Iorio, Fabio Vitali**

Corsi di laurea in Informatica e  
Informatica per il Management  
Alma Mater – Università di Bologna

# Premessa

- Gli studenti di Informatica ricevono una presentazione di progetto simile a questa
- I progetti sono compatibili tra loro ma si differenziano in rapporto ai CFU del corso nel piano di studi
- La regola si applica anche ai corsi mutuati: per un corso da 6CFU fate riferimento a questo progetto, per un corso da 9 al progetto per Informatica
- Gli studenti di Informatica per il Management e i corsi mutuati devono seguire queste specifiche



# Organizzazione dei team

- Ogni persona decide in anticipo se è interessata a sostenere l'esame in estate, autunno, sessione straordinaria o essere ancora indeciso.
- Tutti gli studenti si dividono in team di 2-3 persone. Attenzione: meno di 2 significa troppo lavoro individuale. Più di 3 significa troppo poco. AL MASSIMO 3. NIENTE ECCEZIONI.
- Ogni team porta il progetto insieme (anche qui, non ci sono eccezioni!). Il team dichiara in anticipo la natura del contributo di ciascun membro oppure accetta che chiunque sia interrogato (e nel dettaglio) su tutto il progetto.
- Il sottoscritto NON è coinvolto nell'organizzazione dei team.



# Ruolo di queste specifiche

- Questo documento contiene le specifiche fondamentali del progetto di fine corso.
- **Le specifiche solo volutamente aperte e siete invitati a sperimentare**
- Quanto scritto in nero, salvo esplicite eccezioni, deve essere considerato requisito **OBBLIGATORIO** per la consegna, ma con caratteristiche che ogni gruppo decide in autonomia
  - [Le frasi scritte in arancione e parentesi quadre si riferiscono a servizi opzionali per migliorare la valutazione, e sono **non obbligatorie**.]
  - {Le frasi in verde e parentesi graffe corrispondono a vincoli **obbligatorie** introdotti solo per le esigenze del progetto universitario, non necessari o opportuni in un prodotto vero per il mercato esterno.}
- Se una o più delle specifiche qui introdotte non funzionano, il progetto **NON** è considerato accettabile.
- ***Su Virtuale forniremo eventuali modifiche ai requisiti.***





ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# Animal House

# Animal House\*

Una piattaforma Web  
per curiosi e amanti di  
animali domestici



\* quest'anno le scuse vanno a John Belushi 😊



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA



# Background: [Web] API

- Le API (Application Programming Interface) sono librerie, protocolli e strumenti per permettere di utilizzare gli algoritmi ed i servizi messi a disposizione da un software da parte di un altro software
- Applicazioni e servizi che forniscono una API delegano ad un'applicazione terza aspetti come interazione, interfaccia, navigazione, ecc. e forniscono via API il solo servizio nudo (e i dati)
- Uno degli scopi tipici di ricorso ad API, e il particolare le Web API, è per integrare più servizi in un'applicazione più ricca e potente di quelle utilizzate come base.
- Questo si chiama *mashup* ed è una delle caratteristiche più evidenti di questo periodo storico: mescolare servizi di base per ottenere applicazioni impreviste dai fornitori dei servizi stessi.





# Animal House

- Fate parte di un'azienda che sviluppa software per prodotti e servizi per animali domestici.
- Dovete costruire un'applicazione Web per gestire ed erogare i servizi di AnimalHouse, un'azienda leader di settore
- L'applicazione ha tre facce:
  - Il *game*, ovvero l'applicazione di intrattenimento che permette a curiosi e proprietari di animali di avere informazioni e curiosità sugli animali, fare giochi. Mobile first, solo online, nessun database.
  - Il *front-office*, ovvero l'applicazione per i clienti, mobile first, per permettere le attività di accesso e uso dei servizi online
  - Il *back-office*, ovvero l'applicazione per gli impiegati dell'azienda che abilitano e configurano i diversi servizi







# Il game (1)

Il **game** è un'applicazione a sé stante, orientata a presentare l'azienda, offrire momenti di svago e istruzione ai proprietari di animali

- NON è richiesta autenticazione e i servizi sono disponibili a chiunque si colleghi all'applicazione
- Ha un'ampia disponibilità di semplici giochi a tema animale, e pagine informative e di curiosità specifiche sugli animali dell'utente e sul mondo animale in genere.
- Sfrutta API REST di servizi online, ce ne sono tantissime, alcune nelle prossime slide
- Ha pagine di visualizzazione dei servizi commerciali offerti da Animal House;
- Ha pagine di visualizzazione dei prodotti di e-commerce acquistabili;
- Permette di passare velocemente all'applicazione di front-office

Pensatela come un'app per device mobili e PC. Può essere separata da front-office e back-office o integrata.





# Il game (2)

**Il Game include due servizi e giochi obbligatori (in nero) e almeno due a scelta tra quelli opzionali (in arancione)**

## **Servizi e giochi obbligatori:**

- Immagini o video buffi ed interessanti sul tema (esempio da YouTube)
- **Quiz:** usando ad esempio l'API delle curiosità il sistema genera domande a caso e permette all'utente di giocare

## **Altri possibili servizi:**

- Curiosità sugli animali
- Informazioni utili sanitarie e legali

## **Giochi possibili**

- **Memory:** usando API di immagini il sistema crea coppie di immagini e le dispone nascoste. L'utente deve scoprirle a coppie.
- **Impiccato:** usando API di dizionari di termini sugli animali il sistema presenta una parola complessa che l'utente deve indovinare lettera per lettera
- **Scova le differenze:** il sistema mostra due immagini simili ma con piccoli particolari differenti che l'utente deve scovare (e.g., usare layer SVG per aggiungere i particolari cliccabili)
- [ altri a discrezione del gruppo ]





# Il front-office

Il **front-office** è un'applicazione web tradizionale, solo online, sia per device mobili sia per PC, orientata a fornire accesso ai prodotti e ai servizi offerti dall'azienda.

Necessario fare login per accedere

## Include due servizi obbligatori (in nero) e uno a scelta tra gli opzionali

- **servizi di comunità:** (1) leaderboard con punteggio dei vari giochi del game e (2) bacheca per la condivisione di immagini o aneddoti, **bacheca cerco partner, bacheca aiutatemi, ecc.**
- **e-commerce:** catalogo ragionato e diviso per sezioni di prodotti per animali acquistabili online: cibo, prodotti sanitari, accessoristica, cuccioli.
- **servizi in presenza:** veterinario, dog sitter, toelettatura, pensione estiva, psicologo, visita a domicilio per animali soli, ecc. Ogni servizio prevede una parte di illustrazione del servizio e una parte di accesso e prenotazione (secondo sede e disponibilità)
- **servizi online:** videoconf con l'esperto, con il veterinario, con il proprio animale in ospedale o in pensione, ecc.
- [ altri a discrezione del gruppo ]



# Il back-office

Il *back-office* è la parte dell'applicazione che permette agli amministratori di gestire i dati dei clienti e abilitare e configurare i servizi e i prodotti.

**Include un servizio obbligatorio (anagrafica clienti, in nero) e uno a scelta tra gli opzionali**

Servizi:

- **Anagrafica Clienti:** per gestire le informazioni sui clienti: registrazione, login, cambio password, reset password, cancellazione, preferenze e animali preferiti, punteggi dei giochi
- **Servizi di comunità:** per controllare o cancellare i messaggi nelle bacheche
- **Gestione e-commerce:** per aggiungere e togliere prodotti, prezzi, descrizioni. Ogni prodotto appartiene ad una categoria (e ad una o più sottocategorie) ha sempre un'immagine (con URI online o uploadata sul file system del server)
- **Servizi in presenza o online:** per prenotare servizi in presenza o online, modificare o cancellare prenotazioni, visualizzare disponibilità
- [ configurazione di altri servizi scelti dal gruppo ]



# Animal House: servizi extra

Oltre ai servizi elencati nelle slide precedenti è possibile implementare servizi aggiuntivi come:

- Riconoscimento delle immagini dell'animale dell'utente
- Suggerimenti di estensioni, prodotti aggiuntivi o alternative negli acquisti e nei prodotti
- Recensioni
- etc. ...





# API

- Di seguito un elenco **non vincolante** di API rilevanti per il progetto
- Siete invitati a trovarne e sperimentarne altre
- Elenco API pubbliche: <https://github.com/public-apis/public-apis>
- Fatti e immagini di animali: <https://zoo-animal-api.herokuapp.com/>
- Immagini di cani: <https://dog.ceo/dog-api/>
- Immagini di pesci: <https://www.fishwatch.gov/developers>
- Generatore di Meme: <https://imgflip.com/api>





ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# Regole e consegna



# Valutazione del progetto

- Il progetto è obbligatorio per gli studenti di Informatica per il Management (o Scienze di Internet)
- E' opzionale per gli studenti provenienti da altri corsi di studio
- Il voto complessivo si ottiene dalla media pesata:  
 **$0.7 * \text{voto scritto} + 0.35 * \text{voto progetto}$**
- Per gli studenti che non consegnano il progetto, quindi solo se provengono da altri corsi di studio, vale il voto ottenuto nella prova di laboratorio. La consegna del progetto è comunque molto apprezzata.





# Il lavoro di team

- Tutti i membri dei team sono tenuti a lavorare e lavorare insieme.
- E' meglio essere parte attiva di un progetto mediocre che passiva di un progetto meraviglioso.
- Non saranno tollerati i portatori di pizze
- Mi riservo all'esame di scoprire il contributo individuale di ciascuno, indipendentemente dalla bontà del progetto consegnato
- Tutti i componenti del gruppo devono dare un contributo significativo. Non è sufficiente, ad esempio, lavorare solo a regole CSS
- Tutti i membri del gruppo sono tenuti a conoscere la struttura complessiva del progetto e i costrutti di base usati. Parti specifiche, poi, possono essere discusse solo dalla persona che le ha sviluppate





# Discussione

- La discussione consiste nel mostrare il funzionamento dell'applicazione e il codice sorgente
- Esame in presenza o da remoto. Tutti i membri del gruppo devono essere presenti
- L'applicazione dovrà essere installata su un vostro computer e avere alcuni dati pre-caricati
- E' richiesto avere disposizione un ambiente per aggiornare il codice durante la discussione
- E' ammesso usare soluzioni che richiedono compilazione ma anche in questo vi verrà chiesto di modificare l'applicazione durante la discussione
- La discussione è parte integrante della valutazione del progetto. Non viene valutato solo il codice.





# Consegna

- Il progetto si consegna su Virtuale 5 giorni prima della discussione
- Saranno pubblicati spazi appositi per la consegna ad ogni appello
- Cosa consegnare:
  - Sorgenti
  - README con i componenti del gruppo
  - Istruzioni per installare e usare l'applicazione (possono essere molto brevi, anche una sola riga!)
- Contestualmente alla consegna, iscriversi su AlmaEsami per la corrispondente discussione
  - Iscrivere una sola persona per gruppo





# Vincoli sul progetto: back-end e front-end

- Il focus è su **integrazione di dati** provenienti da API Web e **interazione client-side**
- Il backend è minimale e non è obbligatorio l'uso di un database.
- Il backend essere scritto in un qualunque linguaggio di programmazione ma noi vediamo solo Node/ExpressJS
- La gestione del database è meno importante per gli scopi di questo progetto, rispetto ad integrazione dati, API e mashup
- Il codice client-side può essere scritto in plain Javascript o con JQuery o altri framework (component-based)
- L'uso di un framework non è obbligatorio ma valutato positivamente





# Vincoli sul progetto: persistenza

- E' ammesso salvare i dati direttamente sul filesystem (es. in JSON) o usare database NOSQL o anche SQL-based (**ma che non vedremo**)
- In questo secondo caso usare un servizio online con dati pre-caricati e consegnare un'applicazione già configurata per collegarsi al servizio, senza richiedere l'installazione di un database in locale
- Se i dati sono letti dal filesystem, includere nella consegna i dati minimali per far funzionare l'applicazione:
  - Utenti: amministratore e clienti
  - Altri dati necessari per i servizi implementati





# Altre regole

- Il progetto si consegna e viene valutato una sola volta ma se insufficiente bisogna riconsegnarlo
- Le specifiche del progetto sono valide per questo a.a. (quindi fino a Febbraio 2023). Nel prossimo a.a. il progetto cambierà.
- Il voto resta valido per l'intero anno accademico e per gli anni successivi (almeno finché ci sarò io)
- I voti dei progetti discussi in passato con il Prof. Vitali restano validi per questo anno accademico (almeno)





# Consiglio finale

- Le prove di laboratorio e il progetto sono separati per non imporre vincoli MA è consigliato portarli avanti in parallelo
- Iniziate a lavorare anche al progetto durante il corso  
Lavorare al progetto aiuta a prepararsi alla prova di laboratorio e viceversa
- L'ultima data utile per consegnare questo progetto è febbraio 2023: non arrivate all'ultimo appello!
- Vale anche per la prova di laboratorio 😊





# Domande?



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA