ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Il progetto del corso di Tecnologie Web - A.A. 2022/23

Fabio Vitali, Angelo Di Iorio
Andrea Schimmenti

Corso di laurea in Informatica
Alma Mater – Università di Bologna

Il progetto di fine corso

- Un sistema VERO, che funziona e fa cose utili
- Realizzabile sia in laboratorio che a casa.
- Enfasi in parte sulla programmazione (approccio procedurale) ma soprattutto sui documenti attivi (approccio dichiarativo)
- Enfasi sul mashup di tecnologie esistenti e sofisticate

Avvertenze

- Gli studenti di Informatica per il Management 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 9CFU fate riferimento a questo progetto, per un corso da 6CFU al progetto per Informatica per il Management
- I gruppi di studenti di corsi mutuati possono includere anche studenti di Informatica o Informatica per il Management

Organizzazione dei team

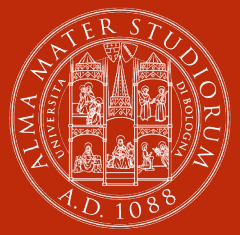
- Ogni persona decide in anticipo se è interessata a sostenere l'esame in estate, autunno, sessione straordinaria o essere ancora indeciso.
- Tutti gli studenti si dividono in team di 2-3 persone. Attenzione: meno di 2 significa troppo lavoro individuale. Più di 3 significa troppo poco. AL MASSIMO 3. 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.

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.

Ruolo di queste specifiche

- Questo documento contiene le specifiche fondamentali del progetto di fine corso.
- Quanto scritto in nero, salvo esplicite eccezioni, deve essere considerato requisito OBBLIGATORIO per la consegna.
  - [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 ***obbligatori*** introdotti solo per le esigenze del progetto universitario, non necessari o opportuni in un prodotto vero per il mercato esterno.}
  - Le frasi in blu sono esempi, e non specifiche di progetto.
- Se una o più delle specifiche qui introdotte non funzionano, il progetto NON è considerato accettabile.
- ***Un'apposita pagina su Virtuale fornirà in maniera sempre aggiornata le eventuali modifiche ai requisiti.***



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Un po' di teoria

Background: strumenti per lo sviluppatore web negli anni '20

Lo sviluppo sw più sorprendente ed evidente degli ultimi anni è la nascita di innumerevoli strumenti per il programmatore, con cui realizzare servizi sofisticati

- Framework: invece di scegliere nuovi linguaggi di programmazione, si usano ora librerie per gli scopi più disparati, facilmente integrabili e mescolabili tra loro
- API (Application Programming Interfaces): invece di sviluppare applicazioni monolitiche che svolgono servizi complessi in un'unica maniera, si forniscono meccanismi di manipolazione delle strutture dati fondamentali e accesso agli algoritmi più sofisticati per applicazioni sviluppate dai clienti.

Questo permette incredibile sofisticazione, grande componibilità, e rapidità di sviluppo precedentemente irraggiungibili.

I framework

- I framework sono librerie che rendono più ricco, sofisticato e semplice l'uso di una tecnologia, come un linguaggio server-side, un linguaggio client-side o le specifiche grafiche di una pagina web.
- Server-side esistono dalla fine degli anni novanta, e hanno reso la programmazione a tre livelli drasticamente più facile.
- Client-side si sono sviluppate a partire dal 2002, su CSS e Javascript, con scopi molto difformi.

Application Programming Interface (API)

- 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, invece che da esseri umani.
- 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.
- Uno degli scopi tipici di ricorso ad 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.

Le API REST

- Le API che ci interessano sono più concretamente quelle che vanno sotto il nome di RESTful API, ovvero che sfruttano al meglio la natura di HTTP e degli URI (i protocolli più importanti del web) per fornire i loro servizi.
- Una API RESTful fornisce:
  - Un URI base a cui accedere per ottenere i servizi
  - Una sintassi degli URI delle entità interrogabili e modificabili
  - Un media type attraverso cui ottenere e fornire dati da utilizzare nei servizi forniti (ad esempio XML, JSON, etc.)
  - Una semantica associata all'uso dei vari verbi HTTP (GET, PUT, POST, DELETE) attraverso i quali attivare e eseguire i servizi offerti.
- La documentazione di un'API tipicamente descrive nel dettaglio nomi (URI), verbi (comandi HTTP) e formati (Internet Media Type) per ogni servizio fornito.

Le API di servizi locali

- Le seconde API che ci interessano sono quelle che permettono alle applicazioni web di accedere a speciali servizi offerti dal device su cui vengono eseguite (tipicamente accesso a periferiche di I/O particolari.
- La diffusione di TCP/IP su device mobili ha ampliato radicalmente la quantità e qualità di periferiche "non tradizionali" a cui l'applicazione può avere accesso:
  - Telecamera (Camera API)
  - Microfono e sintesi vocale (Speech API)
  - Geolocazione (via GPS o altro) (Geolocation API)
  - Suoni (Web Audio API)
  - Vibrazioni (Vibration API)
  - Telefono (Web telephony API),
  - ecc. ecc.
- Ci sono circa 80 API diverse al momento utilizzabili sulla maggior parte dei browser (purché il device offra la funzionalità)

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

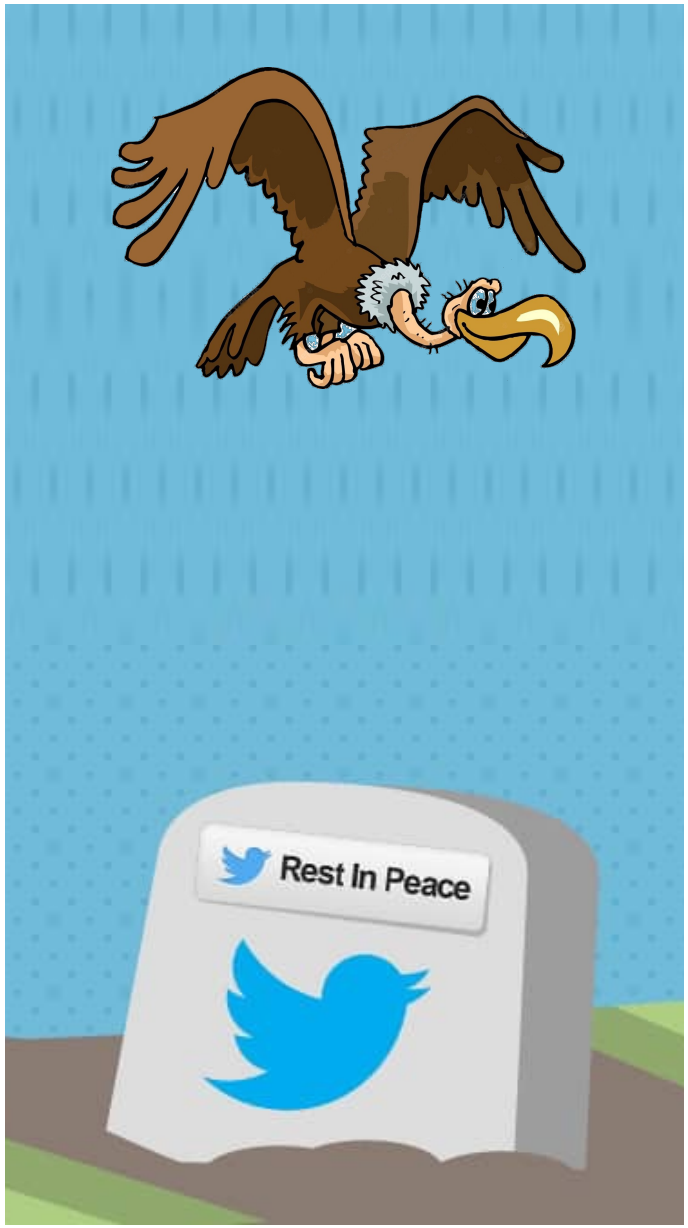
Squealer

Squealer

Banchettare sulle sventure altrui

Un nuovo social,
un po' Twitter,
un po' Tik Tok,
un po' no.

Assunto: Twitter morirà per colpa di scelte sbagliate della nuova gestione.
Si apre dunque uno spazio tecnologico, culturale ed economico enorme da questa sparizione, che Instagram, Telegram e cose del genere non possono colmare da soli.
Squealer andrà a coprire questo vuoto.



Squealer: fondamenti

- Squealer è un social network di brevi messaggi detti *squeal*[1].
- Sebbene non ci sia, come in Twitter, un numero massimo di caratteri nel singolo messaggio, ***c'è un numero massimo di caratteri al giorno, alla settimana e al mese per i messaggi pubblici***
- Un utente può esaurire questa quota in un unico messaggio lungo, pochi messaggi di media lunghezza o tanti messaggi brevi.
- I messaggi possono essere indirizzati ad un singolo utente, oppure ad un canale, oppure a tutti (pubblici). Le reazioni sono importanti
- I messaggi possono contenere testo, link (anche abbreviati), immagini e geolocazioni. Il rapporto tra immagini e testo è che "un'immagine vale mille parole" Per noi ***mille bit,*** cioè 125 caratteri. Una geolocazione viene mostrata come una mappa e conta come un'immagine.

[1] *Squeal* è lo stridio degli avvoltoi

Squealer: la quota di caratteri

- Squealer non ha quota di caratteri per messaggio.
- All'iscrizione, l'utente riceve una quota di default di caratteri che può usare al giorno (D), alla settimana (W < 7*D) e al mese (M < W*4).
- Questa quota è parametrica e deve essere modificabile velocemente.
- La quota residua è sempre visibile e viene automaticamente aggiornata mentre sto scrivendo messaggi.
- Se l'utente ha terminato la quota e sta ancora componendo un messaggio, può usare un certo numero F di caratteri extra. Questa quantità va visualizzata in maniera appropriata, non può essere usata per comporre un messaggio da zero e dovrà essere "pagata" in seguito
- I messaggi ad utenti specifici (non a canali, non pubblici) non usano quota e sono sempre disponibili anche senza quota residua.
- Un utente può aumentare la quota comprandola (per un anno), oppure ottenendo apprezzamenti dal proprio pubblico. Similmente, le reazioni negative diminuiscono la quota (anche quella comprata) fino a farla scomparire (vedi prossime slide)

Squealer: i destinatari

- Squealer usa indirizzi per diffondere gli squeal, ma distingue tra menzioni e destinatari
- La sintassi degli indirizzi è la stessa ma la menzione appare nel corpo del messaggio mentre il destinatario ha un campo tutto suo.
  - @individuo associato ad un singolo utente registrato.
  - §canale (lettere minuscole) è un canale di squeal di proprietà di uno o più utenti, che decidono chi può leggerli e chi può scriverne di nuovi.
  - §CANALE (lettere maiuscole) sono canali riservati a SQUEALER e gestiti dalla redazione. Vedi dettagli nella prossima slide.
  - #keyword sono "canali" estemporanei creati in qualunque momento e accessibili da chiunque senza permessi o limitazioni.

- Gli indirizzi specificati come menzioni sono solo ricercabili, mentre gli indirizzi specificati come destinatari vengono notificati agli iscritti all'indirizzo stesso

Squealer: canali riservati

- Squealer prevede un certo numero di canali riservati (identificati con un §CANALE in lettere maiuscole) e gestiti dalla redazione interna
- Ogni gruppo sceglie quali canali rendere disponibili e la logica di ogni canale. Alcuni possono essere popolati automaticamente sulla base di regole, altri attraverso l'intevento manuale di un moderatore.
- E' obbligatorio includere 3 §CANALI riservati tra cui almeno un canale §CONTROVERSIAL (vedi prossima slide)
- Alcuni esempi:
  - Squeal di tendenza: §TRENDING, §NEWS, §TOP_1000
  - Squeal random: §RANDOM_1000, §RANDOM_ITALY, §RANDOM_BOLOGNA (e qualunque altra area definita dalla redazione)
  - Squeal importanti: §ALL, §EMERGENCY, §EMERGENCY_BOLOGNA, ecc.
  - Squeal controversi: §CONTROVERSIAL_TOP, §CONTROVERSIAL_1000, §CONTROVERSIAL_ITALY, §CONTROVERSIAL_RANDOM
- Alcuni canali non sono mai silenziabili: §ALL, §EMERGENCY, §EMERGENCY_BOLOGNA, ecc.

Squealer: le reazioni

- Di ogni messaggio si contano le *impression* X (il numero di utenti, registrati o meno, che l'hanno visualizzato). Si escludono i destinatari individuali.
- Ogni utente, registrato o meno, può reagire ad un messaggio in maniera positiva o negativa con appositi emoji (scelti dal gruppo, ad es. “Concordo”, “Mi piace”, "sono contrario", "mi disgusta", ecc.)
- Esiste una massa critica (CM) di reazioni positive e negative. Le reazioni polarizzate (R+ e R-) vengono contate separatamente, non si annullano mai. Il valore della massa critica è uguale sia per R+ che R-. Proviamo con CM = 0.25 * X.
- Ogni messaggio che supera la massa critica viene etichettato come "popolare” (R+ > CM), "impopolare" (R- > CM) o "controverso” (se sia R+ che R- superano la soglia CM).
- Un utente che posta messaggi sistematicamente popolari viene premiato con un aumento di quota, se impopolari riceve una diminuzione della quota fino a zero (inclusa la quota acquistata): ogni 10 messaggi con R+>CM vinco 1% della quota iniziale, ogni 3 messaggi con R- > CM perdo 1% della quota iniziale.
- I messaggi controversi non contano per la variazione della quota, ma appaiono nei canali dedicati (§CONTROVERSIAL)

Squealer: il formato dei messaggi

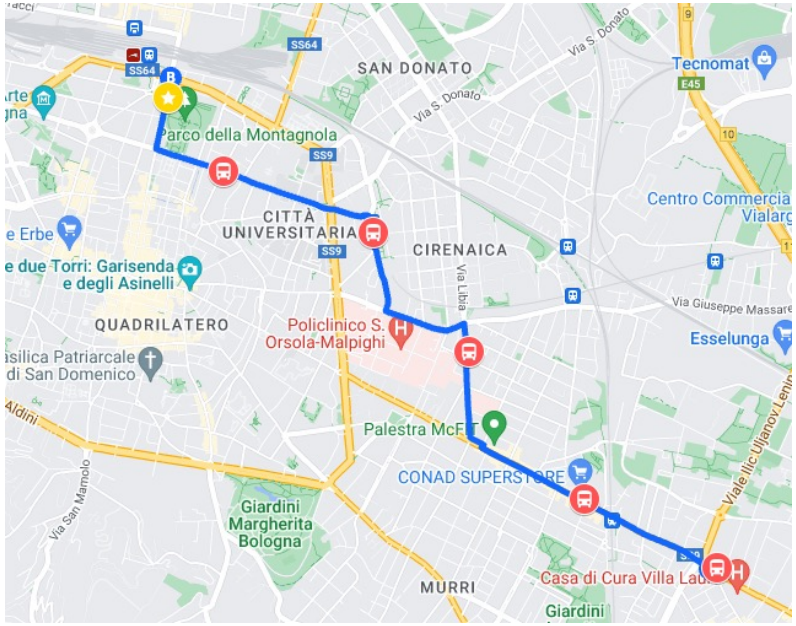
- Ogni messaggio ha un corpo e un elenco di destinatari sotto il controllo dell'autore, e una serie di metadati generati automaticamente o manualmente dal moderatore Squealer.
  - Il corpo è un messaggio di testo, OPPURE un'immagine OPPURE un video, oppure una geolocazione.
  - I destinatari sono un elenco di indirizzi di individui, canali o keyword, senza limiti e senza impatto sulla quota.
  - Data ed ora del messaggio non modificabili
  - # reazioni positive (divise per sottotipo)
  - # reazioni negative (divise per sottotipo)
  - Categoria (privato, pubblico, popolare, impopolare, controverso)
  - Canali Squealer a cui è stato aggiunto dalla redazione
- Esistono inoltre messaggi automatici o derivati da sorgenti esterne (vedi prossima slide)

Squealer: messaggi automatici e da sorgenti esterne

- Ogni gruppo deve implementare almeno 3 tipi di messaggi generati automaticamente. Di questi, il tipo "messaggi temporizzati" è obbligatorio, gli altri 2 a discrezione.
- Alcuni esempi:
  - *Messaggi temporizzati*: generati ogni TOT secondi a contenuto fisso o variabile attraverso data, ora e contatore. Sono utili per servizi di geolocalizzazione (vedi prossime slide). Ad esempio: "Ciao a tutti, questo è il mio messaggio n. {NUM} delle ore {TIME} del giorno {DATE}. ”
  - *News*: news lette da API pubbliche o da feed RSS e trasformati in squeal
  - *Immagini Causali*: immagini recuperate da API pubbliche disponibili su Web e trasformati in squeal
  - *Forse non sapevi che...*: il testo iniziale di pagine random da Wikipedia.
  - *Twitter RIP*: messaggi pubblici letti da canali Twitter attraverso l’API. Sappiamo che Twitter è destinato a scomparire ma gli diamo qualche ultima possibilità ☺
  - …

Squealer: georeferenziazione

- I messaggi temporizzati possono essere usati per costruire dinamicamente mappe e mostrarle in una pagina dedicata.
- Ad esempio, assumiamo che ogni ambulanza, autobus, taxi, camion di corrieri sia dotato di un proprio device con Squealer attivo
- Questo emette ogni N minuti uno squeal di geolocazione sulla propria posizione
- Questi squeal vengono raccolti in un canale e visualizzati come un unico segnale su una mappa, disponibile per tutti gli iscritti a quel canale
- Nella prossima slide alcuni esempi (non vincolanti) di servizi social che è possibile costruire sfruttando questa funzionalità



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Squealer: esempi di servizi social con georeferenziazione

- **Dov'è il mio TAXI?** Alla prenotazione del taxi (ambulanza, ecc.) l'operatore comunica al cliente per telefono una #keyword unica per quella chiamata (ad esempio `#taxiRoma15primula26` che il tassista inserisce tra i destinatari del suo squeal di geolocazione, così il cliente sa sempre da dove arriva il taxi e tra quanto arriva.
- **Dov'è il mio bus?** Al capolinea l'autista della corsa delle 8:12 della linea 27 verso ovest imposta due keyword uniche (ad esempio `#bperbo-Linea27Ovest` e `#bperbo-Linea27Ovest-h08:12`). Gli utenti si possono iscrivere a quelle e ad altre e vedono su una mappa dove sono in ogni momento tutti gli autobus a cui sono iscritti. Quando l'autobus arriva al capolinea e l'autista cambia keyword (`#bperbo-Linea27Est-h08:41`) gli iscritti non ricevono più aggiornamenti.
- **Dove sto andando?** Un influencer gira a piedi per la città e visita un certo numero programmato di negozi. Il suo cellulare emette la posizione associata alla keyword `#LoZioPinoGiraPerBologna`, e i follower che rispondono privatamente indovinando sulla base del percorso quale sarà il prossimo negozio che visiterà vinceranno un buono sconto per l'acquisto in quel negozio.
- **Caccia al tesoro** Un'organizzazione associa ad una keyword degli squeal con indizi di un luogo da indovinare. Chi arriva alla locazione corretta risponde con la geolocazione attuale, e se è esatta riceve degli indizi del luogo successivo. Vince chi arriva per primo all'ultimo luogo.

Squealer: Architettura del sistema

Esistono tre ambienti per usare Squealer:

- La **App**, ovvero l'applicazione per gli utenti finali e altamente mobili. Mobile first, non adatta per grandi volumi di dati e uso professionale.
- Il **SMM dashboard**, ovvero l'applicazione per influencer, VIP e utenti professionali che permette di visualizzare in maniera integrata e complessiva tutte le attività di interazione con il proprio pubblico:, squeal, risposte, reazioni, trend, ecc. Questa è un'applicazione per PC (ma non impossibile da usare su smartphone)
- Il **moderator dashboard**, ovvero l'applicazione per moderatori e redattori gestiti da Squealer, che verificano trend e reazioni, attribuiscono punteggi, risolvono grane, gestiscono blocchi e sblocchi, ecc. Questa è un'applicazione solo per PC.

La app

La app è l'interfaccia principale per i non-professionisti, in cui leggere gli squeal e crearne di nuovi. Ha dunque funzioni di lettura, di scrittura e di accounting.

Funzioni di accounting

- Creazione account, cambio password, reset password, eliminazione.
- Tipo di account: normale, verificato, professional, moderatore squealer.
- Scelta di un social media manager – rimozione del SMM. (entrambi professional: sia utente sia SMM)
- Acquisto caratteri aggiuntivi giornalieri, settimanali, mensili (solo verificati e pro).
- Acquisto di un §canale personalizzato (caratteri minuscoli)
- Aggiunta di altri amministratori al §canale di proprietà

La app

Funzioni di lettura senza login

- Solo messaggi di canali ufficiali Squealer (e.g.: §TRENDING, §NEWS, §TOP_1000, §RANDOM_1000, §RANDOM_ITALY, §RANDOM_BOLOGNA)

Funzioni di lettura con login

- Per default organizzata in maniera temporale inversa, mescolando messaggi personali, canali a cui sono iscritto, canali ufficiali Squealer.
- Si mostrano solo destinatari §canale, §CANALE e #keyword, mai @individui.
- Ricerca per §canale, #keyword e menzione (nel corpo del testo).
- Reazioni: 1 positiva e 1 negativa oppure 3-4 di tipo positivo e 3-4 di tipo negativo.
- Iscrizione e rimozione da §canali e §CANALI a scelta dell'utente.

La app

Funzioni di scrittura

- Creazione nuovo squeal di tipo testo, immagine (copia e incolla da Internet oppure accesso a fotocamera e foto del cellulare) o geolocazione (accesso al sistema di geolocalizzazione del cellulare, visualizzazione della posizione su mappa)
- Risposta a squeal altrui (oltre alla reazione – condivide i destinatari pubblici)
- Counter sempre aggiornati con caratteri residui giornalieri, settimanali e mensili – distinzione tra caratteri effettivamente residui e caratteri che rimarrebbero dopo la pubblicazione di questo post
- Specifica di destinatari (@individui, §canali minuscoli e/o #keyword)
- Ripetizione del messaggio ogni tot secondi (con parti variabili che si aggiornano ogni volta). Beep acustico ad ogni ripetizione.

Lo SMM dashboard

Il ***SMM*** è un'applicazione web tradizionale, solo online, sia per device mobili sia per PC, orientata a fornire accesso all'account di un VIP da parte di un Social Media Manager.

- ***Tipi di account:*** Sia il VIP sia il SMM debbono essere account pro. Il VIP scegli il SMM e può successivamente rimuoverlo.
- ***SMM multiplo***: il SMM può dover gestire diversi account VIP (diciamo anche a 5-6, ma non c'è un numero massimo).
  NON DEVE SBAGLIARSI PER NESSUN MOTIVO.
- ***Scrittura***: il SMM deve poter postare squeal a nome e per conto del VIP. Non deve essere possibile per i destinatari distinguere se lo squeal è del VIP o del suo SMM.
- ***Monitoraggio****:* risposte ai post, post con più e meno reazioni, post popolari, post a rischio controversia o impopolare. Caratteri residui giornalieri, settimanali, mensili
  NON SI PUO' RIMANERE MUTI PER MANCANZA DI CARATTERI
- **Acquisto d'emergenza di caratteri (a prezzo maggiorato)**
- **Trend dei post del VIP***:* grafici con andamenti storici di popolarità, numero di reply, frequenza di post, ecc.
- **Geolocazione fittizia:** il SMM può cliccare in una posizione qualunque della mappa e mandare uno squeal geolocalizzato lì.

Il moderator dashboard

Il *moderator dashboard* è la parte dell'applicazione che permette agli amministratori di Squealer di gestire i dati degli utenti e abilitare e configurare i servizi e i prodotti. Può accedere solo un utente moderatore Squealer.

E' un'applicazione web tradizionale, solo online, principalmente per PC.

- **Utenti.** Il moderatore può elencare gli utenti e filtrarli per nome, tipo e popolarità. Può bloccare e riabilitare gli utenti a mano. Può aumentare a mano i caratteri residui per singoli utenti.
- **Squeal**: il moderatore può elencare i post e filtrarli per mittente, data e destinatari. Può cambiare a mano i destinatari (ad esempio, aggiungere §CANALI ufficiali Squealer). Può cambiare a mano il numero di reazioni positive e/o negative.
- **§canali**: il moderatore può elencare i §canali degli utenti e filtrarli per proprietari, numero di post e popolarità. Può cambiare a mano i proprietari ed il nome. Può bloccare un §canale.
- **§CANALI**: il moderatore può elencare i §CANALI ufficiali Squealer, aggiungerne, toglierne e cambiarne la descrizione (utile per gli altri moderatori). Può aggiungere uno squeal ad un §CANALE o rimuoverlo in qualunque momento. Può aggiungere una regola che attribuisce automaticamente un post ad un canale se soddisfa un criterio.

Requisiti di progetto

- Tutte le parti in nero sono obbligatorie
- Tutte le parti in arancione sono facoltative e generano punteggio extra a discrezione del docente.
- La app è mobile first, è realizzata con il framework Javascript e CSS preferito, ed è pensata soprattutto per essere usata velocemente e facilmente da tutti.
- Il SMM dashboard è sia mobile sia desktop, ed è realizzato con il framework Javascript e CSS preferito, purché diverso da app e da moderator dashboard. E' pensato per un professionista che ha un numero limitato di clienti (1-5) ma che ha un traffico intenso sia in uscita sia di reazioni e like.
- Il moderator dashboard è ***obbligatoriamente*** in Javascript puro (va bene jQuery, va bene un framework CSS a scelta). E' una applicazione web solo desktop.
- Tutti i dati locali vengono memorizzati su un DB Mongo sul server del dipartimento.

Requisiti di progetto

- All'atto della presentazione del progetto tutti i database sono già riempiti con un numero ragionevole di utenti, messaggi, attività, ecc.
- In particolare sono già creati 4 account: "fv", "fvPro", "fvSMM" e "fvMod", con password "12345678". Sono poi anche già creati altri due account Nome Buffo1 e Nome Buffo2, di tipo Pro.
- fvPro, Nome Buffo1 e Nome Buffo2 sono clienti di fvSMM. Nome Buffo 1 è molto popolare ed è prossimo a aumentare la quota di caratteri, Nome Buffo 2 è molto impopolare ed è prossimo a esaurire la quota.
- Tutti e tre questi utenti sono a 50 caratteri dall'esaurire la quota giornaliera (libero settimanale e mensile)
- Ogni utente ha già postato un numero variabile di messaggi (vanno bene anche Lorem Ipsum) che hanno ricevuto un numero libero di like, dislike e risposte.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Valutazione del progetto

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.

Criteri di valutazione (1)

Criteri di valutazione saranno:

1. la generalità dei tool:
   - quanto le soluzioni per la compatibilità sono forzate e quanto sono frutto di scelte ottimali per framework, organizzazione del codice e uso corretto delle tecnologie disponibili
2. la flessibilità:
   - quanto le soluzioni tecniche adottate sono solide, strutturate, facilmente comprensibili, facilmente estendibili, facilmente adattabili a nuovi device / browser / sistemi operativi / modelli di dati / modelli di annotazione

Criteri di valutazione (2)

3. l'usabilità:
   - Quanta attenzione è data alle esigenze di utenti (sia giocatori, sia clienti, sia dipendenti che usano il back-office) che non conoscono i dettagli del modello di applicazione utilizzata.
4. la sofisticazione grafica
   - Quanta attenzione viene data alla presentazione delle informazioni, al rapporto tra dimensioni delle maschere e dimensioni dei dati da rappresentare, al rapporto tra label comprensibili e dati formalizzati, alla corretta differenziazione nei tipi di dati e di annotazioni.
5. l'accessibilità
   - Tutte le applicazioni sono accessibili e compatibili con gli standard WCAG e ARIA

Vincoli hard

1. Le tre applicazioni sono fatte con tre modelli applicativi diversi: il back-office ***deve*** essere realizzato con vanilla Javascript. Le altre applicazioni con due framework diversi tra Angular, React, Vue, Svelte, etc.
2. Il framework per la grafica è libero ma mi aspetto sofisticazione grafica, facilità d'uso e accessibilità. Vanno bene Bootstrap, Tailwind, Foundation, ma anche altri a vostra scelta purché compatibili con gli altri vincoli.
3. Il deploy ***deve*** avvenire su uno/due container docker sulle macchine del dipartimento.
4. Ogni applicazione che usi forme di packing o compilazione (inclusi webpack e typescript) ***deve*** avere i sorgenti caricati nella directory *source* dello spazio web)
5. Tutte le applicazioni devono essere accessibili.
6. Tutti i database vengono presentati già popolati.

Criteri di valutazione (3)

- Lo scritto pesa l'70% del voto finale
- Il progetto base pesa il 35% del voto finale
- L'aggiunta di funzionalità facoltative porta il peso fino al 40% del voto finale, a discrezione del docente.
- In casi eccezionali (tutte le funzionalità facoltative, servizi di AI integrati, ulteriori funzionalità non previste in queste specifiche) si può arrivare fino al 45% del voto finale, a discrezione del docente.

Il contributo individuale

- Ogni membro di ogni team deve dimostrare di aver contribuito in maniera determinante alla realizzazione del progetto.
- Ad inizio della presentazione ogni membro dichiara che cosa ha realizzato, e il docente, in totale autonomia, decide se questo contributo è o non è sufficiente.
- Realizzare solo HTML e CSS non è sufficiente.
- Realizzare parti marginali del codice (login, logout, lettura delle preferenze, ecc.) non è sufficiente
- Un candidato ideale si è occupato sia della parte HTML/CSS, sia della parte di programmazione, sia client sia server.
- La distribuzione ideale dei compiti è funzionale e non architetturale.

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.

Suggerimenti per l'esame

- Venite alla presentazione con il progetto che funziona. Se non va io vi faccio tornare. Per questo preferisco
  - vedervi una settimana dopo l'appello con il progetto che funziona
- piuttosto che
  - perdere tempo con la presentazione di un progetto che non va,
  - mandarvi via in lacrime, e
  - vedervi una settimana dopo l'appello con il progetto che funziona
- Venite allo scritto avendolo preparato. Non c'è niente di più irritante di vedere ragazzi svegli e competenti (vi si riconosce) che prendono 8 o 10 allo scritto perché ci hanno solo provato.
- Lo scritto non è difficile per chi ha studiato, è impossibile per chi non l'ha fatto.

Attenzione all'appello di febbraio

- Quanto detto NON si applica all'appello di febbraio.
- Il ~~28~~ 29 febbraio 2024 si conclude la possibilità di presentare il progetto di quest'anno.
- Non riducetevi all'ultimo, cercate di portare il progetto negli appelli estivi ed autunnali
- Gli slot a disposizione per presentare il progetto a febbraio non sono infiniti. Tutti gli anni questi slot (più del doppio degli altri appelli) si esauriscono molto presto.
- ***Garantisco slot di presentazione solo se fate richiesta entro il 31/12/2023. Non accetto prenotazioni dopo il 31/1/2024.***
- Io cerco di ***non*** essere più esigente a febbraio, ma forse potrò dedicare meno attenzione al vostro meraviglioso progetto.
- Non riducetevi all'ultimo (l'ho già detto, lo so)

Flessibilità del corso

- Prova scritta e prova di progetto sono indipendenti.
  - Il progetto è sempre di gruppo
  - Lo scritto è sempre individuale
- Potete provare lo scritto tutte le volte che volete
  - Il voto precedente verrà cancellato solo se consegnate un nuovo scritto
  - Gli scritti sono solo alle date degli appelli ufficiali
- Potete presentare il progetto tutte le volte che volete
  - Solo se lo decide consensualmente TUTTO IL GRUPPO
  - Potete ritirarvi dalla presentazione del progetto in qualunque momento e tornare una settimana o due dopo con le correzioni che ritenete opportune.

Rigidità del corso
(nessuna eccezione per nessun motivo)

- **Il progetto deve funzionare**. Completamente ed esattamente secondo specifiche. Nel corso del tempo queste specifiche possono anche evolvere.
- **Il progetto deve risiedere su una macchina del dipartimento**. Questo include SIA il codice SIA tutti i dati del progetto (tranne quelli per cui è ***esplicitamente*** previsto l'uso di un server condiviso)
- **Potete installare librerie e SW per il progetto** a vostro piacimento, MA verificate prima che sia eseguibile sulle macchine e sui sistemi operativi offerti dal dipartimento e con i permessi d'uso di un utente normale,
- **Il progetto deve venire presentato da tutto il gruppo insieme**, in presenza oppure online su MS Teams. In nessun caso è accettabile che si presenti in una data una parte del gruppo e in una data diversa il resto del gruppo.

Attenzione!

Vi ho già detto di non ridurvi all'ultimo?

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

**Fabio Vitali, Angelo Di Iorio,
Andrea Schimmenti**

Dipartimento di Informatica – Scienze e Ingegneria
Alma mater – Università di Bologna

www.unibo.it