

Oggetto dell'incontro – Discussione del capitolato con Zero12

<i>Partecipanti</i>	<i>Durata presenza</i>
Marco Bernardi	40 minuti
Gabriel Rovesti	40 minuti
Mircea Plamadeala	40 minuti
Andrea Meneghello	40 minuti
Davide Milan	40 minuti
Davide Sgrazutti	40 minuti
Milo Spadotto	40 minuti

Ora inizio: 14:00

Ora fine: 14.40

Presente da parte dell'azienda Alex Massaro, rappresentante di Zero12.

Questi i chiarimenti alle nostre domande, presentate come si vede:

- 1) Maggiori dettagli/chiarimenti sul prodotto che vogliono
- 2) Le tecnologie consigliate sono obbligatorie o possiamo gestirci come vogliamo (frontend/backend)?
- 3) Dubbi in merito alle librerie da usare (web-mobile)
- 4) Ambiente unico/standardizzato per non avere problemi di compatibilità
- 5) Tempistiche incontro e se forniscono materiali
- 6) Formazione AWS corsi

Centro del progetto sono le API e la gestione del backoffice.

Riguardo le tecnologie, sono dei consigli implementativi, non stretti.

Per la realizzazione del DB è dividere lo schema client dalla realizzazione:

una parte di backoffice (sito), da parte degli admin/tenant cioè aziende e le aziende di traduzione.

Le API vengono salvate direttamente; non serve avere più di un database, poiché ne basta uno singolo e le API permettono di interagire, funzionando in push e in pull.

API per POST, GET, DELETE con il GET da parte dei client.

La login viene salvata dal database e la componente del client la usa per fornire il servizio di traduzione.

Il discorso multi-tenant serve per far vedere le traduzioni ai singoli clienti, senza che si vedano tutti tra di loro; ci sarà una sorta di superadmin che vede tutto e le persone all'interno di una company vedono solo i loro elementi.

Il superadmin crea l'utente X per la company A. Quando l'utente si logga (servizi di Amazon già pronti, es. servizio che fa l'autenticazione che usa un token salvato nei cookies). Viene salvato in REST e con le API possiamo capire subito l'utente della company A, restituendo le traduzioni della company di interesse.

I corsi saranno di sessioni di un paio d'ore e lo stack tecnologico si basa principalmente su Amazon. In generale, danno tre tecnologie per studiare quello che serve. Anche l'utilizzo della libreria serve per permettere l'utilizzo del servizio.

L'idea è di avere una libreria *wrapper* che permette di avere direttamente di avere informazione sulle traduzioni (lato client); mentre le librerie Kotlin/Swift è per app Android/iOS.

Scegliamo una sola → se vogliamo la parte frontend, per dire, usiamo solo quella.

Servirà una demo (es. pagina/app), anche senza grafica, con le stringhe tradotte. Nella parte backoffice, possiamo usare dei template bootstrap per la realizzazione, usando tranquillamente cose pronte.

Serve da noi avere un feedback sulle varie tecnologie utilizzate, andando verso un linguaggio o un framework X; vorrebbero avere un resoconto (1 paginetta), capendo per i singoli servizi usati se ci siamo trovati bene o altro, capendo bene il tutto.

Per la formazione, sulla base delle tecnologie consigliate, danno cose standard e usate da tutti (giustamente). Ci danno della documentazione sulle cose da fare, affinché il gruppo in 1-2 settimane capisce se fa per lui. Ci si può trovare un'ora/due per parlare del backend/frontend o cose del genere. Non si è obbligati a fare le cose sempre uguali, andando sul sicuro. La parte più ostica è AWS, per configurare le routing delle IP da raggiungere.

Viene definito un canale Slack di chat e per le SAL di avanzamento. Ogni volta che ci vogliamo sentire, usiamo quel canale per comunicare e parlare insieme tutti, sia per gli incontri.

Potrebbe succedere, sulla base del numero dei gruppi, fare delle sessioni tutti insieme (se si ha una tecnologia comune); non per forza deve essere in presenza, alcuni da remoto e alcuni in sede.

Non serve fare entrambe, ma possiamo scegliere se sito o app.

In generale, i più partecipativi sono stati Davide M. e Marco, dando spunti di discussione e domande. L'incontro è stato positivo e ha convinto tutti della scelta fatta nel capitolato.