**Descrizione del contesto:**

È stata richiesta la realizzazione di un sistema per il tracciamento e visualizzazione della posizione di atleti durante allenamenti o gare. Gli atleti hanno una gestione del profilo e la possibilità di creare nuovi allenamenti o di iscriversi ad una gara, mentre gli organizzatori posono creare gare o vedere anche in tempo reale i dati telemetrici di un atleta. Nelle specifiche è previsto che l’applicazione per il runner sia diversa da quella per l’organizzatore.

**Descrizione architetturale:**

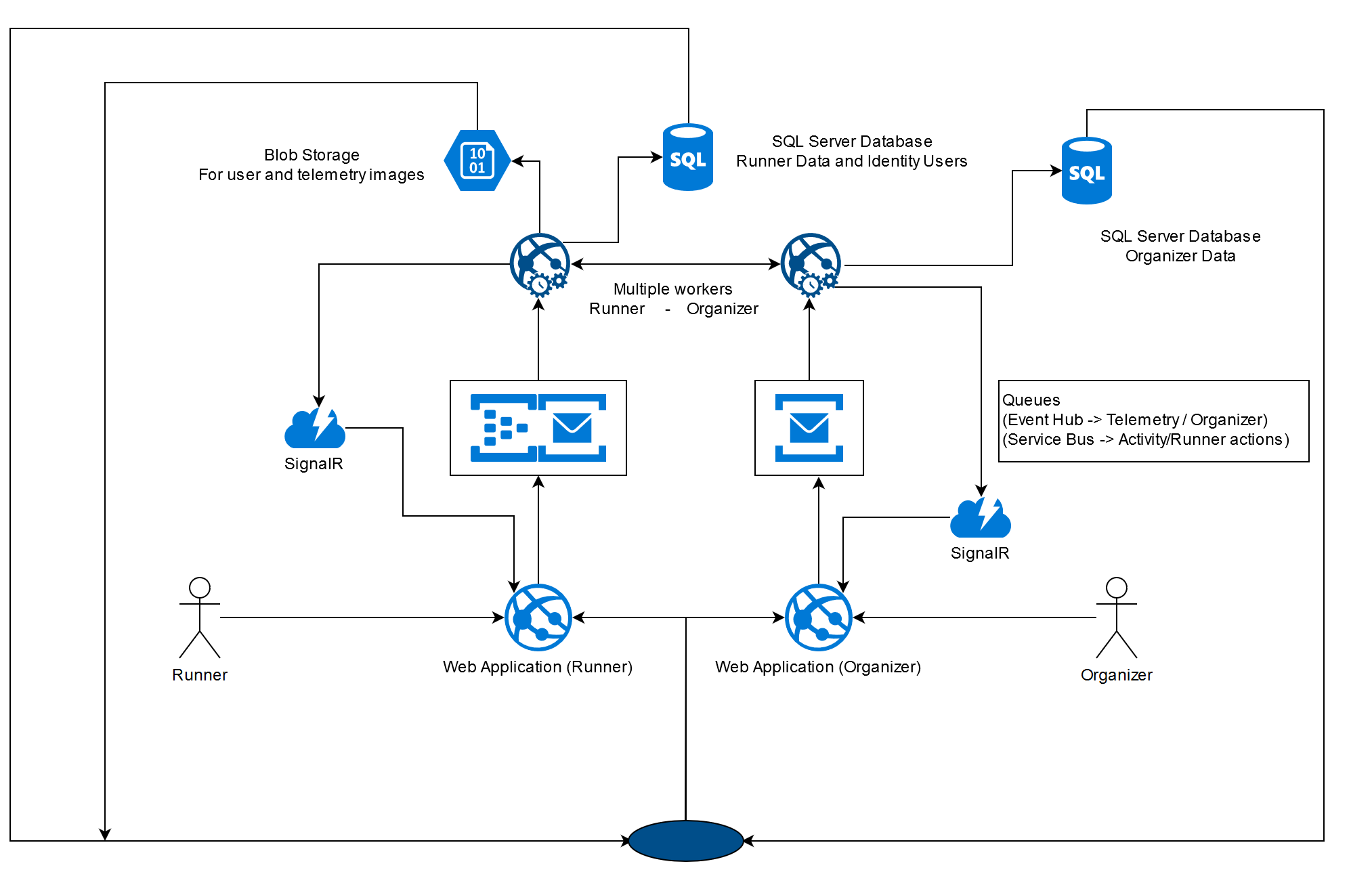


Figura - Schema dell'architettura

L’applicazione è scritta interamente in C# e il sistema gira su Azure.

La parte di front end è formata da un progetto per il runner e uno per l’organizzatore. Ogni progetto ha un suo database per i dati telemetrici / delle attività, ma i dati utente di Entity Framework (per l’autenticazione) sono salvati in un database singolo. Entrambi i progetti fanno riferimento ad un sistema di code. Per il runner, le richieste vengono messe in coda tramite Event Hub per l’invio dati telemetrici, che è stato scelto per la capacità di supportare code grandi e numerose richieste contemporanee. Per altri tipi di scritture al database viene utilizzata una coda di Service (es: inserimento attività) con prestazioni inferiori ma che comunque garantisce una disassociazione tra database e web app. Per l’organizzatore invece viene usato solo Event Hub in endpoint separati.

Sono presenti due worker (uno per web app) che si occupano di aggiornare il loro database specifico e, se necessario, comunicare tra loro per inviare dati all’altro database. Inoltre, il worker per il runner può inserire immagini nel blob storage, o inviare un segnale di redirect tramite SignalR, usato per evitare di mostrare dati non aggiornati.

Da notare che i dati vengono ricevuti direttamente in query di lettura, i quali possono contenere URL ad immagini presenti nel blob storage; è solo la scrittura a mettere i dati in coda.

**Descrizione funzionalità:**

Dalla parte del runner, è possibile

* Gestire le proprie attività
* Iscriversi a gare
* Modificare l’immagine di profilo
* Correre

L’applicazione richiede di essere autenticati per poter usarla. Non sono presenti pagine per utenti non loggati in quanto le telemetrie vengono distinte da codici utente.

Dalla lista delle attività è possibile selezionare di visualizzare l’attività, o aprirla/chiuderla se si hanno i permessi necessari, ovvero se si è l’autore dell’attività. Dopo aver selezionato la voce ‘visualizza’, è possibile visualizzare dettagli sull’attività e la mappa, che è stata realizzata tramite l’uso delle api di Google Maps.

Inizialmente vengono caricate le tappe percorse precedentemente, che vengono mostrate tramite marker. I marker possono essere selezionati per visualizzare informazioni aggiuntive; inoltre le tappe sono attraversate da linee per indicare il percorso eseguito. Quando queste vengono caricate, ogni 5 secondi viene effettuata una richiesta ajax per inviare i dati di geolocalizzazione (possibile tramite funzionalità browser). Questa passa per l’Event Hub, il worker, a viene inserita nel database.

È inoltre possibile scattare una foto, che viene inserita assieme alla richiesta successiva. Questa viene salvata in uno storage blob di Azure, e viene mostrata quando viene selezionato un marker, se una foto è associata al dato.

Dalla parte dell’organizzatore, è possibile:

* Gestire e creare le proprie gare
* Visualizzare dati telemetrici di utenti che percorrono la gara.

All’organizzatore è infatti permesso di visualizzare in tempo reale i dati telemetrici degli utenti. Questa funzionalità è permessa tramite la combinazione di Event Hub (che permette di avere più consumer per eseguire subcribe multipli) e SignalR (per effettuare lo streaming di dati).