**2. System design**

Il modello architetturale scelto è stato il pattern **MVC**, il quale è più di un modello architettonico che utilizza il design per far rispettare il suo schema di alto livello ed è più gestibile in quanto diviso in parti.

Abbiamo usato il pattern **MVC** perché:

* È più facile riutilizzare il codice, quindi lo sviluppo è più veloce
* Il codice è più organizzato, quindi è più facile da capire e mantenere
* È più facile testare il codice.
* è più sicuro

Il pattern architetturale, è diviso in tre parti:

1. **Model**

* Incapsula lo stato dell’applicazione
* Deve permettere di accedere ai dati
* Deve notificare i cambiamenti di stato

1. **View**

* Mostra il modello
* Gestisce l’interazione con l’utente

1. **Controller**

* Rappresenta la logica applicativa
* Collega le azioni dell’utente con modifiche dello stato
* Sceglie cosa essere mostrato

