

# CODEBUSTERS

Progetto: *HD Viz*  
[codebusterswe@gmail.com](mailto:codebusterswe@gmail.com)

## Verbale Interno del 10-03-2021

### Informazioni sul documento

<b>Versione</b>	1.0.0-0.3
<b>Approvatori</b>	Rago Alessandro
<b>Redattori</b>	Baldisseri Michele
<b>Verificatori</b>	Scialpi Paolo
<b>Uso</b>	Interno <i>Zucchetti</i>
<b>Distribuzione</b>	Prof. Vardanega Tullio Prof. Cardin Riccardo Gruppo <i>CodeBusters</i>

### Descrizione

Documento riassuntivo della riunione svolta in data 10-03-2021

## Registro delle modifiche

Versione	Data	Nominativo	Ruolo	Descrizione
1.0.0-0.3	11-03-2021	Rago Alessandro	Approvatore	Approvazione del documento
0.1.0-0.3	11-03-2021	Safdari Hossain	Verificatore	Revisione del documento
0.0.1-0.3	10-03-2021	Baldisseri Michele, Sialpi Paolo	Redattore, Verificatore	Stesura del verbale e verifica

# 1 Informazioni generali

- **Motivo della riunione:**
  - Valutazione periodo precedente alla RP;
  - Scelta dell'architettura del prodotto;
  - Riutilizzo delle funzionalità del PoC;
  - Divisione del lavoro nei prossimi giorni.
- **Luogo riunione:** videoconferenza tramite server Discord<sup>G</sup>.
- **Data:** 10-03-2021;
- **Orario d'inizio:** 10:00;
- **Orario di fine:** 12:30;
- **Partecipanti:**
  - Baldisseri Michele
  - Sassaro Giacomo
  - Scialpi Paolo
  - Safdari Hossain
  - Pirolo Alessandro
  - Zenere Marco
  - Rago Alessandro

## 2 Resoconto

- **Valutazioni sul lavoro svolto fino al periodo appena concluso:** il gruppo in seguito alla realizzazione del PoC e alla consegna della documentazione in RP si è considerato soddisfatto del lavoro svolto, specialmente per il fatto di essere riuscito a consegnare tutto in tempo rispetto a quanto dichiarato nel *Piano di Progetto*. Il rispetto delle scadenze prefissate ha portato sicuramente buon umore e voglia nel proseguire di questo passo con la successiva fase di progettazione di dettaglio e codifica.
- **Scelta del design architetturale:** il secondo punto affrontato è stato quello di determinare quale fosse l'architettura migliore da utilizzare per l'applicazione. In seguito a un lungo confronto è stato scelto come design architetturale il *Model-View-ViewModel (MVVM)*, design derivante dal più comune Model-View-Controller (MVC).  
I motivi principali che hanno portato il gruppo ad optare per questa architettura rispetto all'MVC sono stati:
  - MVVM permette una più forte separazione tra business logic e presentational logic. La vista infatti non è legata in alcun modo al modello, ma è il view-model che mette in comunicazione le due parti;
  - L'utilizzo di unico Controller (come accade nell'MVC) per gestire l'intera vista è risultato sfavorevole per due motivi:
    - \* In questo caso il controller dovrebbe essere il componente padre di tutta la vista. Questo porterebbe che un singolo cambiamento dello stato del controller causi la ri-renderizzazione di tutti i componenti della vista, con conseguenti rallentamenti nel normale utilizzo della web application;
    - \* Avere l'intera application logic in unico file porterebbe gravi difficoltà di cooperazione tra i membri del gruppo nella codifica della vista. I componenti in React hanno infatti spesso una logica interna, che in questo caso dovrebbe essere spostata nell'unico controller. Così facendo se due membri lavorano contemporaneamente a due componenti della vista che richiedono una logica interna, entrambi dovranno modificare il file in cui è contenuto il controller, con conseguenti conflitti.
  - In seguito a ricerche è risultato come MVVM sia un design architetturale molto diffuso per le web application realizzate con React per la maggiore facilità di implementazione.
- **Riutilizzo del PoC:** il gruppo ha valutato positivamente la possibilità di riutilizzare alcune delle funzionalità già presenti nel PoC per velocizzare la fase di codifica;
- **Suddivisione del lavoro:** per concludere il gruppo si è diviso il lavoro da svolgere nei prossimi giorni, con l'obiettivo di ottenere un'architettura stabile per l'applicazione e un primo esempio di vista.

### 3 Registro delle decisioni

Codice	Decisione
Interno-10.1	<i>Model-View-ViewModel(MVVM)</i> scelto come design architetturale per <i>HDViz</i> .
Interno-10.2	Riutilizzo delle funzionalità principali già implementate nel <i>Proof of Concept</i> .
Interno-10.3	Creazione di un'architettura stabile basata su MVVM per l'applicazione e una bozza della vista nella settimana seguente.

Tabella 1: Decisioni della riunione interna del 10-03-2021