



ProApes

proapes11@gmail.com

Verbale Interno 2020-04-02

Versione	1.0.0-1.10
Data approvazione	2020-04-02
Responsabile	Fiammetta Cannavò
Redattori	Valentina Signor
Verificatori	Igor Biolcati Rinaldi
Stato	Approvato
Lista distribuzione	<i>ProApes</i> <i>Prof. Tullio Vardanega</i> <i>Prof. Riccardo Cardin</i>
Uso	Interno

Sommario

Resoconto dell'incontro interno effettuato dal gruppo *ProApes* il giorno 2020-04-02
mediante l'uso di *Discord*

Diario delle Modifiche

Versione	Data	Modifica	Autore	Ruolo
v1.0.0-0.3	2020-04-02	<i>Approvazione del documento</i>	Fiammetta Cannavò	<i>Responsabile di Progetto</i>
v0.1.0-0.3	2020-04-02	<i>Revisione complessiva di coerenza e coesione (Verificatore: Igor Biolcati Rinaldi)</i>	Valentina Signor	<i>Progettista</i>
v0.0.1-0.3	2020-04-02	<i>Stesura del documento (Verificatore: Igor Biolcati Rinaldi)</i>	Valentina Signor	<i>Progettista</i>

Indice

1	Informazioni generali	3
2	Ordine del giorno	3
3	Resoconto	4
3.1	Discussione riguardo il colloquio <i>Agile</i> per la <i>Product Baseline</i>	4
3.2	Stesura e-mail da inviare al <i>Prof. Riccardo Cardin</i>	4
3.3	Approvazione schema di versione complessivo per i prodotti finali . .	6
4	Riepilogo delle decisioni	7

1 Informazioni generali

- **Luogo:** - ;
- **Strumento:** *Discord*;
- **Data:** 2020-04-02;
- **Ora inizio:** 10:00;
- **Ora fine:** 12:30;
- **Segretario:** Valentina Signor;
- **Partecipanti:**
 - Francesco Bari;
 - Igor Biolcati Rinaldi;
 - Fiammetta Cannavò;
 - Federico Carboni;
 - Alessandro Discalzi;
 - Giacomo Piran;
 - Valentina Signor.

2 Ordine del giorno

- Discussione riguardo il colloquio *Agile* per la *Product Baseline*;
- Stesura e-mail da inviare al *Prof. Riccardo Cardin*;
- Approvazione schema di versione complessivo per i prodotti finali.

3 Resoconto

3.1 Discussione riguardo il colloquio *Agile* per la *Product Baseline*

Grazie ad uno scambio di mail avvenute nei giorni precedenti con il *Prof. Riccardo Cardin*, attraverso l'uso della mail ufficiale, si è stabilita la data in cui il gruppo sosterrà il colloquio *Agile*: venerdì 2020-04-10 nello slot 13:55-14:10.

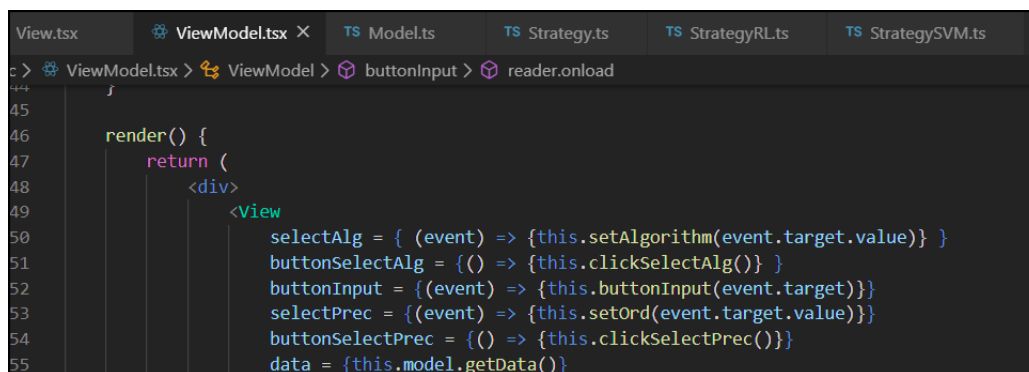
Durante l'incontro di oggi si sono trattati alcuni aspetti riguardanti il materiale da presentare e le modalità di ripartizione del lavoro durante questo periodo. In particolare si sono prese alcune decisioni per delineare quale fosse la strategia migliore da adottare per affrontare la *Product Baseline*; si è poi provveduto a dettagliare meglio la struttura e i contenuti dell'*Allegato Tecnico*. Tale documento formale esterno, sarà inviato al committente la mattina stessa del colloquio *Agile* attribuendo alle scelte architetture del gruppo concrete basi documentali. In merito a ciò si è evidenziato che:

- lo stile redazionale impiegato cercherà di non essere troppo prolisso o troppo schematico; verrà fornita prima una descrizione generale dell'architettura del prototipo realizzato, che poi si articolerà nei rispettivi modulo esterno ed interno;
- verranno inseriti i *diagrammi delle classi*, *diagrammi di sequenza* e *diagrammi dei package*, nonché l'indicazione dei *design pattern* notevoli impiegati;
- si includeranno tabelle e areogrammi così da documentare precisamente quali tra i requisiti siano quelli soddisfatti e quali no, ponendo particolare attenzione a quelli obbligatori.

3.2 Stesura e-mail da inviare al *Prof. Riccardo Cardin*

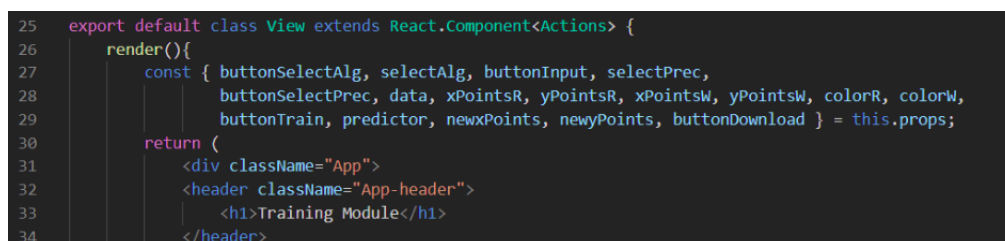
Il gruppo ha discusso quali fossero i *design pattern* più adatti da applicare per definire l'architettura del prodotto software che il team sta cercando di realizzare. A seguito dell'insorgere di alcuni dubbi in merito, i membri hanno provveduto a stendere una e-mail da inviare entro l'indomani al *Prof. Riccardo Cardin*. In essa si è cercato di spiegare nel dettaglio il perché delle decisioni scelte in modo da sanare per tempo eventuali dubbi ed incomprensioni; i punti trattati sono stati i seguenti:

- dubbio rispetto l'utilizzo del *design pattern Strategy* con un'interfaccia, implementata nelle due strategie concrete per *RL* e per *SVM*, e potenzialmente estendibile ad altre librerie di *Machine Learning*. A tal proposito viene esposta al committente l'intenzione, sorta a seguito di una serie di discussioni e studi, di effettuare una variazione alla struttura originale del *design pattern* in questione. Di seguito se ne riporta parte del testo: "*Inoltre abbiamo pensato di eliminare la classe Context ritenuta superflua in quanto agiva solo da ulteriore interfaccia verso il modello, senza alcuna funzionalità aggiuntiva: la creazione della strategia è stata inclusa nel modello - È ragionevole eliminarla in questo modo per semplificare la struttura del programma? Se sì, in questo caso è ancora possibile parlare di Strategy?*";
- decisione d'implementare il *design pattern MVVM* ritenendola l'architettura migliore da impiegare in *React*. Il gruppo evidenzia come per il passaggio di informazioni si sfrutterebbero le proprietà delle componenti in modo da legare il ViewModel alla Vista e rendere quest'ultima indipendente da qualsiasi logica:



```
ViewModel.tsx > ViewModel > buttonInput > reader.onload
45
46 render() {
47   return (
48     <div>
49       <View
50         selectAlg = { (event) => {this.setAlgorithm(event.target.value)} }
51         buttonSelectAlg = {() => {this.clickSelectAlg()} }
52         buttonInput = {(event) => {this.buttonInput(event.target)}}
53         selectPrec = {(event) => {this.setOrd(event.target.value)}}
54         buttonSelectPrec = {() => {this.clickSelectPrec()}}
55         data = {this.model.getData()}
```

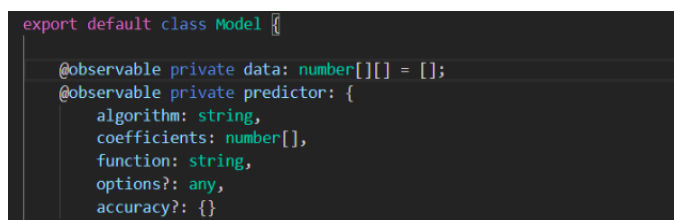
Figura 1: Codice del ViewModel



```
25 export default class View extends React.Component<Actions> {
26   render(){
27     const { buttonSelectAlg, selectAlg, buttonInput, selectPrec,
28             buttonSelectPrec, data, xPointsR, yPointsR, xPointsW, yPointsW, colorR, colorW,
29             buttonTrain, predictor, newXPoints, newYPoints, buttonDownload } = this.props;
30     return (
31       <div className="App">
32         <header className="App-header">
33           <h1>Training Module</h1>
34         </header>
```

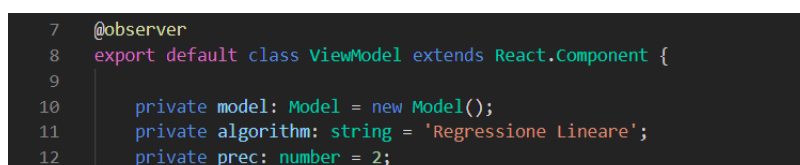
Figura 2: Codice della Vista

- Individuazione di un metodo che assicuri un aggiornamento continuo della Vista rispetto al Modello; nel dettaglio: *"Abbiamo scelto di utilizzare MobX (<https://mobx.js.org/README.html#introduction>): una libreria di state management, spesso utilizzata assieme a React, che permette di implementare l'idea del pattern observer attraverso i decorator @observable e @observer - In questo modo il View-Model potrà aggiornare la Vista quando necessario".* Di seguito il gruppo ha anche fornito le immagini relative al Modello e al ViewModel:



```
export default class Model {
  @observable private data: number[][] = [];
  @observable private predictor: {
    algorithm: string,
    coefficients: number[],
    function: string,
    options?: any,
    accuracy?: {}
  };
```

Figura 3: Codice del Modello



```
7 @observer
8 export default class ViewModel extends React.Component {
9
10   private model: Model = new Model();
11   private algorithm: string = 'Regressione Lineare';
12   private prec: number = 2;
```

Figura 4: Codice del ViewModel

- dubbio riguardo l'applicazione della medesima architettura trattata al punto precedente fatta eccezione di MobX, anche per il modulo interno. A supporto della sua tesi il team evidenzia come Grafana presenti già un proprio sistema di refresh delle componenti con un intervallo di tempo configurabile da interfaccia utente.

3.3 Approvazione schema di versione complessivo per i prodotti finali

Il gruppo ha ripreso a discutere del numero di versione (*Verbale Interno 2020-03-23*), cercando ora di individuare un codice unitario che fosse applicabile sia alla documentazione che al software. Ciascun membro del team ha potuto portare le sue idee e perplessità in merito grazie a cui si è giunti alla conclusione che la numerazione sarebbe stata composta da:

$$[x].[y].[z]-[a].[b]$$

dove

- **x.y.z**: conferma quanto già stabilito in precedenza;
- **a.b**: cerca di evidenziare come documentazione e componenti software concorrono entrambi alla composizione del prodotto nel suo insieme. Essi vengono dunque visti non come componenti separate ma come un'unica componente singola; nel dettaglio:
 - **a**: indicherà una versione completa e funzionante del prodotto software (requisiti obbligatori, metriche, test soddisfatti ecc.) e di tutti i documenti richiesti dai committenti;
 - **b**: indicherà l'incremento stabilito nel *Piano di Progetto* di cui soddisfa gli obiettivi.

4 Riepilogo delle decisioni

Codice	Decisione
VI_2020-04-02.1	Prese decisioni riguardanti il colloquio <i>Agile</i> che il gruppo <i>ProApes</i> sosterrà venerdì 2020-04-10 nello slot 13.55-14:10.
VI_2020-04-02.2	Redazione e-mail da inviare al <i>Prof. Riccardo Cardin</i> per renderlo partecipe delle decisioni del gruppo inerenti l'architettura e i <i>design pattern</i> che verranno applicati. In tal modo si cerca di sanare prontamente eventuali incomprensioni fra le parti.
VI_2020-04-02.3	Stabilita seconda parte del codice di versione per garantire omogeneità fra documenti e parte software.

Tabella 2: Riepilogo delle decisioni