Diario di lavoro

|  |  |
| --- | --- |
| Luogo | Trevano |
| Data | 23.01.2019 |

|  |
| --- |
| Lavori svolti |
| Oggi mi son dedicato completamente alla pagina di login e registrazione.  Implementato il template indicato la scorsa volta, ho avuto non pochi problemi nel fare in modo che me lo leggesse. Questo perché inizialmente avevo semplicemente copiato il codice che mostravano, senza registrarmi per scaricarlo. Dopo qualche tentativo andato a vuoto tuttavia, mi son reso conto di come il gioco non valesse la candela e che pur di evitare di perdere ulteriore tempo valeva la pena registrarsi e scaricarlo. Una volta fatto, i miei problemi, dopo aver cancellato comunque la funzione che dava problemi nel file JS, sono stati risolti. La funzione cancellata non aveva un vero scopo, o quanto meno non gliel’ho trovato. Cancellandola ho comunque verificato che tutte le funzioni primarie fossero quanto meno attive.  Mentre risistemavo il sito adattandolo alle mie esigenze, ho notato come delle icone che mi dava a disposizione mancasse quella per un calendario. Così ho deciso di fare un salto sulla pagina (<https://codyhouse.co/> ) che metteva a disposizione le immagini presenti all’interno dei campi input.  Con mia sorpresa, ho scoperto un ottimo sito, che sicuramente mi sarà utilissimo da oggi in avanti. Contiene infatti una buona quantità di template e componenti (sebbene questi non siano modificabili secondo licenza) implementati con una grafica leggera e con sapiente utilizzo di JavaScript, il tutto per rendere il sito ancora più fruibile e user-friendly. È un framework, ma ogni pagina è disponibile presso il loro GitHub, e quindi accessibile e opensource.  Ho scelto i template per alcune delle prossime pagine, ovvero:  Pagina di inserimento informazioni personali: <https://codyhouse.co/demo/breadcrumbs-multi-steps-indicator/index.html#0> per i passaggi e questo (<https://codyhouse.co/demo/contact-form/index.html> ) per l’inserimento dei dati.  Pagina di filtraggio dati e possibilità di stampa: <https://codyhouse.co/gem/content-filter>  Inoltre per l’amministratore, che avrà a disposizione molte più funzioni quali le modifiche alle pagine, traduzioni e aggiunta di parametri, pensavo ad un sistema simile <https://codyhouse.co/gem/stretchy-navigation> riguardo alla scelta.  Alla fine ho deciso di utilizzare FontAwesome per le icone, poiché pratico nel suo utilizzo e già conosciuto e utilizzato da me per altri siti.  Sono quindi tornato alla pagina, terminando con successo quanto concerne la grafica di questa prima parte, e implementando già l’inizio della logica, creando la classe User (parametri presenti: nome, cognome, data di nascita, email, genere e password). Il genere, dopo averci riflettuto un po’, ho deciso di definirlo come booleano. Lo 0 corrisponde al genere femminile, e l’1 al maschile. Questo mi permette di avere molta più libertà, anche riguardanti l’utilizzo di lingue diverse.  Iniziando ad implementare il codice, mi son reso conto del primo vero problema: avendo una chiave primaria composta da 3 attributi, il sistema da me conosciuto per usare Spring Security diventa inefficace, quanto meno senza le dovute modifiche. Ho quindi cominciato a ricercare una possibile soluzione. Baeldung, uno dei miei siti più fidati riguardanti la programmazione in Spring, mi ha dato questa soluzione <https://www.baeldung.com/spring-security-extra-login-fields>. Tuttavia, è un sistema molto diverso rispetto a quello che uso io, e nel corso della prossima lezione andrà rivisto. |

|  |
| --- |
| Problemi riscontrati e soluzioni adottate |
| Iniziale difficoltà nell’implementare correttamente il template all’interno del progetto.  Ho notato un problema di gestione, in quanto al momento Spring Security l’ho configurata in questo modo:    L’ultima funzione è quella principale, che controlla il sistema e gestisce le richieste. Verrà poi descritto accuratamente all’interno della documentazione.  Il problema sussiste col fatto che, come si può notare, i parametri utilizzati sia in quel metodo che nel primo sono 2, email (come username) e password. Ora invece la mia chiave primaria è composta da ben 3 parametri, e non essendoci la possibilità di utilizzare altri parametri devo valutare meglio la situazione e, eventualmente, sostituire il sistema di gestione Spring Security (esistono infatti moltissimi standard per esso, questo che uso al momento è semplicemente quello con cui mi son trovato meglio per lavorare). |

|  |
| --- |
| Punto della situazione rispetto alla pianificazione |
| Dao e Mapper, non avendo ancora creato la vera logica dietro, non sono stati creati. Il resto è stato recuperato. Questa è dovuto più ad un mio errore di valutazione nella pianificazione, in quanto ha molto più senso al momento creare prima tutta la struttura per registrazione e login e solo in seguito pensare allo sviluppo delle restanti funzioni. |

|  |
| --- |
| Programma di massima per la prossima giornata di lavoro |
| Finire la pagina di registrazione/login con logica implementata. |