

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



Migrazione a dispositivo mobile di una applicazione web tramite tecnologia Ionic

Tesi di laurea

Relatore

Prof. Tullio Vardanega

Laureando

Enrico Buratto

ANNO ACCADEMICO 2019-2020

Sommario

Il seguente documento descrive il tirocinio che ho svolto presso la sede di Padova dell'azienda **Sync Lab s.r.l.**, d'ora in poi "*Sync Lab*" o "*l'azienda*", nel periodo che va dal 31/08/2020 al 23/10/2020. Lo stage ha avuto una durata complessiva di circa **300 ore**, ed è stato supervisionato dal mio tutor esterno, Andrea Giunta, e dal mio relatore, prof. Tullio Vardanega.

Scopo dello stage è stato studiare e utilizzare il *framework Ionic*, tecnologia che permette la realizzazione di applicazioni *mobile* ibride a partire da codice *JavaScript* o *TypeScript*, per il **porting** di un applicativo web preesistente su dispositivi mobili.

La migrazione verso dispositivi mobili che ho effettuato si è basata sulla componente *front-end* dell'applicativo *SyncTrace*, sviluppata da altri studenti durante i precedenti tirocini e completata ufficialmente il 28/08/2020; questa componente è stata sviluppata tramite il *framework Angular* in accoppiata al linguaggio *TypeScript*.

L'**obiettivo** di questo applicativo è inquadrabile nell'ambito del *contact tracing* per la prevenzione di infezione da *Sars-CoV2*: esso permette infatti il monitoraggio di persone infette, e a rischio infezione, da parte di medici ed esercenti.

Il presente documento è suddiviso nei seguenti capitoli:

- * **Capitolo 1**: presentazione dell'azienda, comprendente i processi messi in atto dall'azienda per soddisfare il cliente, la metodologia di sviluppo, le tecnologie di interesse e un approfondimento sulla propensione dell'azienda all'innovazione;
- * **Capitolo 2**: descrizione della proposta di *stage*, comprendente gli obiettivi e i vincoli a cui mi sono dovuto attenere e la motivazione della scelta. Descrive inoltre la fase iniziale del mio tirocinio, corrispondente alla formazione sulle tecnologie e sul precedente progetto, punto di partenza del mio lavoro;
- * **Capitolo 3**: descrizione dettagliata del progetto di *stage*, ossia delle attività di sviluppo, verifica, validazione e collaudo che ho effettuato per ottenere il prodotto finale;
- * **Capitolo 4**: valutazione retrospettiva dello *stage*, contenente il soddisfacimento degli obiettivi e le competenze professionali che ho maturato.

In questo documento ho utilizzato le seguenti **convenzioni tipografiche**:

- * I termini in lingua straniera e i termini propri sono evidenziati dall'utilizzo del corsivo;
- * Piccole sezioni di testo possono essere riportate in grassetto, al fine di evidenziarne l'importanza nel discorso;
- * Tutte le tabelle e le immagini riportano un numero progressivo, una descrizione e una fonte.

*“Boschi ed acque, venti ed alberi, saggezza, forza e cortesia, che il favore della giungla
ti accompagni”*

— Rudyard Kipling

Ringraziamenti

Padova, Dicembre 2020

Enrico Buratto

Indice

1	Contesto aziendale	1
1.1	Introduzione	1
1.2	L'azienda	2
1.3	Processi aziendali	4
1.3.1	Processi	4
1.3.2	Metodologia Agile	5
1.4	Tecnologie utilizzate	6
1.5	Propensione all'innovazione	9
2	Il progetto nel contesto aziendale	13
2.1	Il rapporto tra stage e azienda	13
2.2	L'azienda in relazione al contesto attuale	14
2.3	Lo scopo dello stage	15
2.4	Vincoli e obiettivi dello stage	17
2.4.1	Vincoli temporali	17
2.4.2	Vincoli organizzativi	18
2.4.3	Vincoli tecnologici	18
2.4.4	Obiettivi	19
2.5	Motivazione della scelta	20
2.6	Formazione	21
2.6.1	Tecnologie	22
2.6.2	Progetto	23
3	Il progetto di stage	25
3.1	Analisi dei requisiti	25
3.2	Progettazione	25
3.3	Codifica	25
3.4	Verifica	25
3.4.1	Analisi statica	25
3.4.2	Test di unità	25
3.5	Validazione e collaudo	26
3.6	Risultati ottenuti	26
4	Valutazioni retrospettive	27
4.1	Soddisfacimento degli obiettivi	27
4.2	Bilancio formativo	27
4.2.1	Maturazione professionale	27
4.2.2	Rapporto tra università e lavoro	27

Elenco delle figure

1.1	Rappresentazione grafica dello <i>smart working</i>	1
1.2	Logo di Sync Lab s.r.l..	2
1.3	Moduli di SynClinic.	3
1.4	Funzionalità del progetto Seastream.	3
1.5	Metodologia Scrum.	5
1.6	Panoramica delle tecnologie utilizzate da Sync Lab.	6
1.7	Architettura dello <i>Spring framework</i>	8
1.8	Pattern MVVM, adottato anche dal <i>framework Angular</i>	8
1.9	Funzionamento del <i>framework Electron</i> con <i>NodeJS</i>	9
1.10	Alcune università con le quali l'azienda collabora.	10
1.11	I progetti di ricerca e sviluppo a cui Sync Lab collabora.	10
2.1	<i>Academia vs business - xkcd</i>	13
2.2	<i>Dashboard</i> di COVID-19 nel mondo.	14
2.3	Rappresentazione schematica del <i>contact tracing</i>	15
2.4	Schermata iniziale dell'applicazione <i>SyncTrace</i>	16
2.5	Schermata iniziale della <i>webapp</i> per, rispettivamente, medici ed esercenti.	16
2.6	Porzione del registro delle attività.	18
2.7	Logo di <i>StageIT 2020</i>	20
2.8	<i>Homepage</i> della piattaforma <i>Udemy</i>	21

Elenco delle tabelle

2.1	Suddivisione settimanale delle attività	17
2.2	Obiettivi di progetto originali.	20
2.3	Ulteriore obiettivo, aggiunto in corso di progetto.	20

Capitolo 1

Contesto aziendale

1.1 Introduzione

In questo documento descrivo la mia esperienza di *stage* presso la sede di Padova di **Sync Lab**, un'azienda nata come *software house* e che, negli anni, si è affermata nel campo dell'*ICT* (*Information and Communication Technology*).

È di particolare rilievo, nella mia particolare esperienza, il periodo storico in cui questo tirocinio è avvenuto: ho infatti effettuato lo stage tra il mese di settembre e il mese di ottobre 2020, ossia durante l'emergenza sanitaria globale dovuta alla malattia *COVID-19*. La pandemia in atto ha influenzato radicalmente il lavoro di molte persone, me compreso: al fine di ridurre le possibilità di contagio, infatti, ho dovuto svolgere gran parte del tirocinio in regime di *smart working*, potendo accedere alla struttura solo una volta alla settimana, e alle infrastrutture quasi esclusivamente in maniera remota.



Figura 1.1: Rappresentazione grafica dello *smart working*.

Fonte: corriere.it

Nonostante la situazione non favorevole, ho avuto comunque modo di interfacciarmi con i dipendenti dell'azienda; sono quindi riuscito anche a raccogliere le informazioni

riguardanti l'azienda che, unite a delle ricerche online, ho riportato in questo capitolo.

1.2 L'azienda

Sync Lab nasce come *software house* a Napoli nel 2002; negli anni, l'azienda si espande a gran velocità, aprendo sedi a Roma, Milano, Padova e Verona. Al giorno d'oggi, l'azienda conta 5 sedi, per un totale di oltre 250 dipendenti e più di 150 clienti diretti e finali.

Dall'apertura ad oggi, Sync Lab si è tramutata in un *system integrator* grazie alla maturazione delle competenze tecniche e metodologiche in ambito software. Un tratto distintivo dell'azienda è la grande attenzione posta alla gestione delle **risorse umane**: testimonianza di ciò è il basso *turn-over*, segno che i collaboratori condividono un progetto comune e concreto.

Altro segno d'eccellenza sono le certificazioni di qualità che l'azienda ha conseguito; finora, infatti, l'azienda ha ottenuto le certificazioni per gli standard *ISO 9001*, *ISO 14001*, *ISO 27001* e *ISO 45001*.



Figura 1.2: Logo di Sync Lab s.r.l..

Fonte: syncclab.it

Prodotti e servizi offerti

Una certa attenzione è posta dall'azienda alla diversificazione dei prodotti e dei servizi offerti; essi sono infatti inquadrabili in diverse aree tematiche, quali salute, privacy, sicurezza, telecomunicazioni, finanza, territorio e ambiente.

Alcuni dei prodotti che l'azienda offre al momento sono i seguenti:

- * **DPS 4.0:** software per la gestione degli adempimenti alla *Privacy GDPR - General Data Protection Regulation*, utilizzato da svariate aziende per attuare correttamente quanto previsto da tale regolamento europeo; tale prodotto permette di censire, tracciare e controllare chi può trattare dati personali in azienda;
- * **StreamLog:** sistema finalizzato al soddisfacimento dei requisiti fissati dal *Garante per la Protezione dei dati personali*, utilizzato dagli amministratori di sistema per controllare gli accessi agli utenti al fine di soddisfare i requisiti di *privacy* richiesti dal garante;
- * **SynClinic:** sistema informativo per la gestione integrata dei processi, clinici e amministrativi, di ospedali, cliniche e case di cura. Questo applicativo fornisce svariate funzionalità, che intersecano i bisogni del personale amministrativo e quelli del personale clinico delle strutture sanitarie, permettendo di gestire e monitorare tutte le fasi del percorso di cura del paziente. È utilizzabile sia in *cloud* che *on premises*;

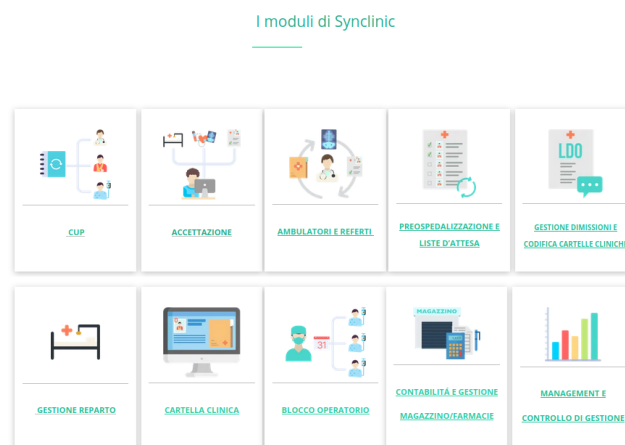


Figura 1.3: Moduli di SynClinic.

Fonte: synclinic.it

- * **StreamCrusher:** tecnologia che aiuta le aziende a effettuare corrette decisioni di *business*, a identificare tempestivamente eventuali criticità e a riorganizzare i processi in base a nuove esigenze. Questo software è in grado di raccogliere, indicizzare e interpretare i dati, siano essi *log* di applicazione o di sistema, *alert*, dati di configurazione o modifiche ai sistemi, al fine di estrapolarne informazioni utili all'*IT management*;
- * **Wave:** software che si propone come integrazione tra il mondo della videosorveglianza e quello dei *Sistemi Informativi Territoriali*, permettendo di avere una visione geo-referenziata della distribuzione delle telecamere installate sul territorio e consentendo così all'utilizzatore di avere un impatto visivo immediato sull'area di copertura di una data installazione reale, o di avere un'anteprima di tale area in fase di progettazione;
- * **Seastream:** piattaforma pensata per migliorare l'efficienza, la sicurezza e il processo di innovazione del settore marittimo; per fare ciò l'azienda fornisce, attraverso questa piattaforma, un *Fleet Operation Center (FOC)*, ovvero un sistema di monitoraggio avanzato delle flotte armatoriali operative in tutto il mondo, e un *Harbor Operation Platform (HOC)*, ovvero una piattaforma di servizi per gli operatori portuali.

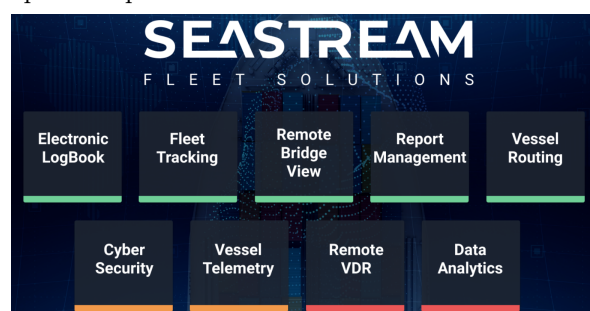


Figura 1.4: Funzionalità del progetto Seastream.

Fonte: synclab.it

Clienti principali

Sync Lab collabora con numerose aziende italiane e multinazionali, sia pubbliche che private. Tra le aziende **private** più importanti possiamo trovare *Sky*, *Eni*, *Enel*, *Vodafone*, *Accenture*, *Fastweb*, *Tim*, *UniCredit* e *H&M*.

Tra le collaborazioni con **enti statali** e parastatali, invece, troviamo quelle con *Trenitalia*, *RAI*, *Poste Italiane*, la *Regione Lazio* e il *Ministero dell'Economia e delle Finanze*.

1.3 Processi aziendali

1.3.1 Processi

L'azienda persegue i propri obiettivi attuando i processi qui elencati.

Consulenza

L'azienda fornisce servizi di consulenza informatica a svariate imprese, sia pubbliche che private; questi servizi hanno lo scopo di far evolvere, sia in termini di sviluppo che di competitività, i clienti dell'azienda. Per fare ciò, Sync Lab collabora con altre aziende di consulenza e con specialisti del settore.

Fornitura

Il processo di fornitura viene istanziato ogniqualvolta un cliente assume Sync Lab per lo sviluppo e la realizzazione di un prodotto. Contemporaneamente alla realizzazione, l'azienda svolge delle attività che possano migliorare questo processo. In particolare:

- * **Qualità del software:** il software viene controllato e ottimizzato, anche attraverso l'utilizzo delle *best practice*, al fine di aderire alle regole aziendali;
- * **Verifica delle procedure:** le procedure vengono verificate, al fine di poter agire in maniera correttiva nel caso in cui si dovessero manifestare dei problemi;
- * **Analisi e miglioramento degli standard:** gli standard aziendali vengono analizzati e, possibilmente, migliorati; conseguenza di ciò è un costante miglioramento anche dal punto di vista di qualità del software.

Sviluppo

Per quanto riguarda il processo di sviluppo, Sync Lab fa uso della metodologia *Agile*, descritta più dettagliatamente in seguito. Questa permette di coinvolgere il cliente durante tutto il processo, tenendolo aggiornato sull'evoluzione dello sviluppo del prodotto e ricevendo di ritorno le sue opinioni, al fine di poter agire sia correttivamente che migliorativamente.

Manutenzione

Una volta consegnato il prodotto, l'azienda assicura le attività di manutenzione per tutto il ciclo di vita del software. La manutenzione offerta dall'azienda è di tre tipi:

- * **Correttiva**, corrispondente alla correzione di eventuali difetti;

- * **Adattiva**, ossia il riadattamento del software a nuovi requisiti quali l'ambiente di produzione o l'architettura;
- * **Evolutiva**, ossia l'aggiunta o l'aggiornamento in senso migliorativo di porzioni di software.

1.3.2 Metodologia Agile

Per il processo di sviluppo, l'azienda fa uso di una metodologia *Agile* che molto si avvicina al modello *Scrum*. Punto cardine del metodo di sviluppo di Sync Lab, infatti, è la continua interazione con gli *stakeholder*, ossia i clienti: questi vengono coinvolti durante tutto il processo, venendo aggiornati sull'evoluzione del prodotto; questo permette all'azienda di ricevere *feedback* che possono aiutare a migliorare il prodotto e ad adattarlo al meglio alle esigenze.

Come ho potuto constatare di persona durante il mio periodo di tirocinio, il modello adottato dall'azienda prevede uno sviluppo che procede per *sprint*, ossia un'unità di base di durata fissa compresa tra una e quattro settimane a seconda degli obiettivi posti. A ogni *sprint* corrisponde una funzionalità; questa viene verificata insieme al cliente, per tastarne la soddisfazione o ricevere consigli che possano migliorare tale nuova funzionalità.

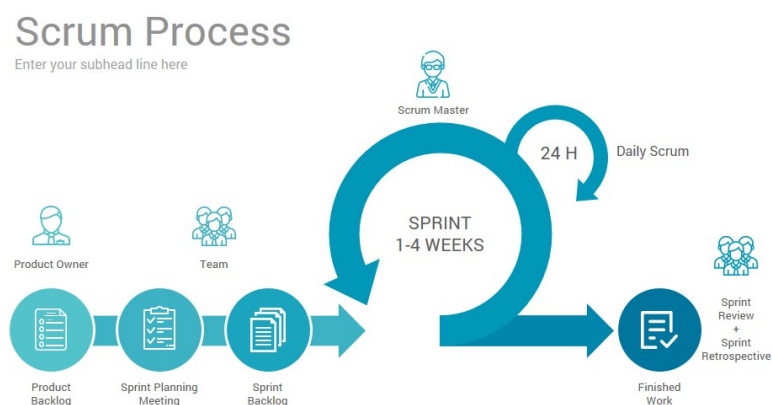


Figura 1.5: Metodologia Scrum.

Fonte: antevenio.com

Lo sviluppo di un prodotto in Sync Lab passa attraverso cinque attività, inquadrabili tutte in quanto previsto dalla metodologia *Scrum*.

La prima attività che viene svolta è la redazione di una lista di cose da fare per portare a termine il progetto; nel caso del mio progetto di *stage*, ad esempio, ho definito insieme al tutor aziendale le *feature* che avrei dovuto implementare, e abbiamo visionato assieme i *bug* presenti nel codice che avrei dovuto utilizzare come *baseline*, al fine di sapere cosa avrei dovuto correggere di quanto già fatto. Nella metodologia *Scrum*, questa lista assume il nome di **Product Backlog**.

Dopo aver redatto questa lista, solitamente viene realizzata una pianificazione degli *sprint* che è necessario effettuare per portare a termine il progetto. In ambito *Scrum* questa pianificazione viene chiamata **Sprint Planning**, e nel caso del mio tirocinio è corrisposta alla divisione in incrementi che avrei dovuto effettuare per completare

il piano di lavoro. Successivamente a questo, vengono individuati dei sottoinsiemi di obiettivi per ogni *sprint* definito durante l'attività chiamata **Sprint Backlog**. Un esempio pratico di questa attività è la definizione degli incrementi che ho effettuato prima di cominciare lo *stage*.

Una volta definiti tutti gli obiettivi e aver correttamente diviso le tempistiche in **Sprint**, questi vengono ad uno ad uno eseguiti; nell'ambito del mio *stage*, questo è corrisposto allo sviluppo di quanto previsto dal piano di progetto. All'esecuzione di ogni incremento, inoltre, di questo è stata verificata l'aderenza agli obiettivi posti, che nella metodologia *Scrum* vengono chiamati **Sprint Goal**.

Durante il mio *stage* ho inoltre potuto sperimentare anche due degli eventi facenti parte della metodologia *Scrum*, ossia il *Daily Scrum* e lo *Sprint Review*:

- * Il **Daily Scrum**, come definito dai creatori di questa metodologia, è un evento a cadenza giornaliera in cui i membri del gruppo di lavoro discutono dell'andamento dello *sprint*. Questo evento viene svolto dai diversi *team* di lavoro dell'azienda giornalmente; nel caso specifico del mio tirocinio, essendosi questo svolto in gran parte in *remote working*, questo evento non ha potuto avere cadenza giornaliera. Nonostante questo, ho effettuato svariati allineamenti con il tutor aziendale, e questo si può almeno in parte configurare con quanto previsto dalla metodologia;
- * Lo **Sprint Review**, ossia la verifica di quanto effettuato durante uno *sprint* alla fine di questo, viene effettuato da tutti i gruppi di lavoro al raggiungimento degli obiettivi fissati dallo *sprint*. Nel caso specifico del mio tirocinio, questo evento ha avuto luogo quasi settimanalmente, con la verifica di quanto fatto durante l'incremento definito.

1.4 Tecnologie utilizzate

Sync Lab fa uso di diversi linguaggi di programmazione, *framework* e strumenti di supporto moderni e funzionali, al fine di soddisfare i clienti; oltre a questo, l'azienda è costantemente aggiornata sulle tecnologie di riferimento e pronta ad espandere le proprie conoscenze con le tecnologie più moderne.

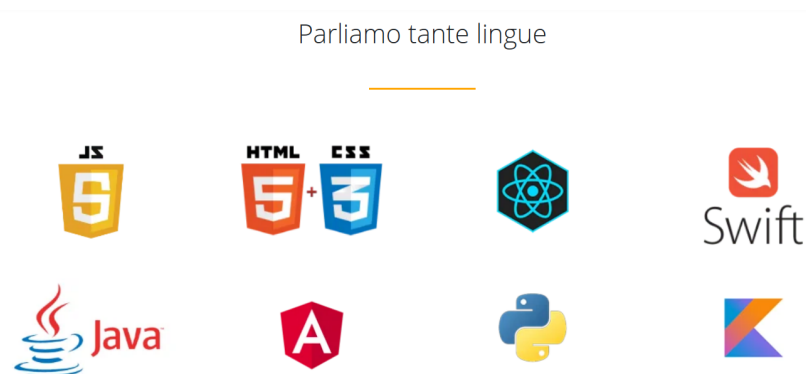


Figura 1.6: Panoramica delle tecnologie utilizzate da Sync Lab.

Fonte: synclab.it

Tra i linguaggi di programmazione più utilizzati troviamo i seguenti:

- * **Java:** linguaggio di programmazione ad alto livello orientato agli oggetti, largamente utilizzato dalle aziende; in particolare, *Java* viene utilizzato da Sync Lab, in accoppiata al *framework Spring*, per lo sviluppo dei servizi *REST* necessari alle componenti *back-end* di svariati applicativi, tra cui ad esempio *SynClinic* e *StreamCrusher*. Questo linguaggio, abbinato al *framework Spring*, è stato utilizzato anche per lo sviluppo della componente *back-end* dell'applicativo *SyncTrace*, oggetto del mio tirocinio;
- * **JavaScript:** linguaggio di programmazione orientato agli oggetti e agli eventi, originariamente pensato per la creazione di effetti dinamici interattivi per i siti web ma sempre più utilizzato come linguaggio *general purpose* per lo sviluppo di applicativi web e non web. Viene utilizzato dall'azienda per realizzare la componente logica delle *Single Page Application*;
- * **TypeScript:** *super-set* di *JavaScript* sviluppato da *Microsoft*. Questo linguaggio estende la sintassi di *JavaScript* in modo che ogni programma scritto in *JavaScript* possa funzionare anche con *TypeScript*, e come il primo viene utilizzato dall'azienda per realizzare la componente logica delle *Single Page Application*. Si può trovare questo linguaggio, abbinato al *framework Angular*, in svariati prodotti dell'azienda, tra cui *SynClinic* e *DPS4.0*; è inoltre il linguaggio con il quale è stata scritta la componente *front-end* della *web application* di *SyncTrace*;
- * **HTML5 e CSS3:** linguaggi di *markup* utilizzati, anche dall'azienda, per modellare la componente visiva delle *Single Page Application* e, più in generale, dei siti web. Questi linguaggi di *markup* sono usati dall'azienda principalmente nei progetti che coinvolgono il *framework Angular*, come ad esempio *SynClinic* e *DPS4.0*;
- * **Kotlin:** linguaggio di programmazione *general purpose* e multi-paradigma, sviluppato da *JetBrains*, utilizzato dall'azienda per sviluppare le applicazioni mobili per i dispositivi *Android*. Un esempio di utilizzo di questo linguaggio è l'applicazione mobile per il *contact tracing* del progetto *SyncTrace*;
- * **Swift:** linguaggio di programmazione orientato agli oggetti, sviluppato da *Apple* e utilizzato dall'azienda per sviluppare le applicazioni mobili per i dispositivi *iOS*.

L'azienda utilizza anche svariati *framework* a supporto della programmazione; alcuni tra i *framework* più utilizzati dall'azienda sono:

- * **Spring:** *framework open-source* per lo sviluppo di applicazioni con linguaggio di programmazione *Java*; viene utilizzato, possibilmente combinando il *core* del *framework* con altri progetti quali *Spring Boot* e *Spring Data*, per sviluppare applicativi lato *server*. Esempio di utilizzo di questo *framework* da parte di Sync Lab sono svariate applicazioni web la cui componente *back-end* è sviluppata con l'ausilio di queste tecnologie, come ad esempio *SynClinic* e *StreamCrusher*;

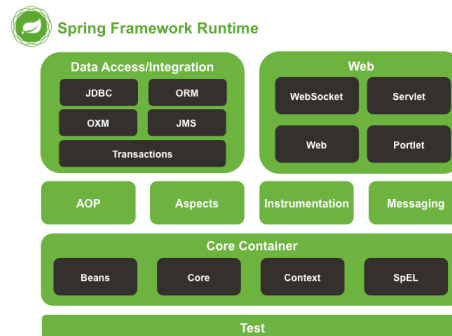


Figura 1.7: Architettura dello *Spring framework*.

Fonte: javaboss.it

- * **Angular:** *framework open-source* per lo sviluppo di applicazioni web tramite linguaggio di programmazione *TypeScript*. L'utilizzo principale di questa tecnologia risiede nello sviluppo di *Single Page Application* reattive e costruite su un *back-end* composto da servizi *REST*. Esempi di utilizzo di questo *framework* sono *SynClinic* e *DPS4.0*, già citati in precedenza.

Oltre alla possibilità di sviluppare applicativi veloci e funzionali, questo *framework* offre anche un *design pattern* di tipo *Model-View-ViewModel* nativo, fattore che facilita la progettazione e lo sviluppo delle applicazioni;

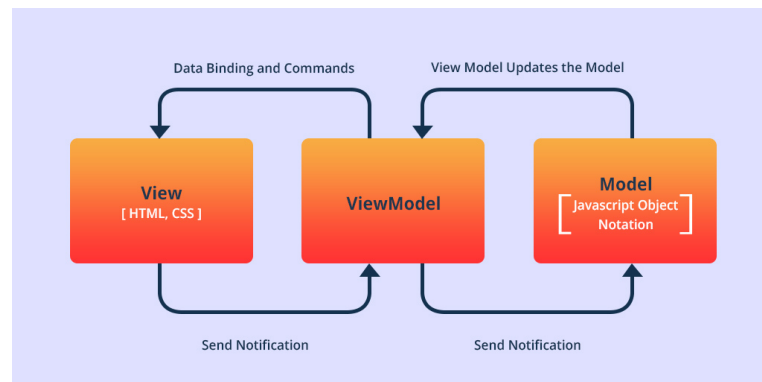


Figura 1.8: Pattern MVVM, adottato anche dal *framework Angular*.

Fonte: alphalogicinc.com

- * **Electron:** *framework open-source* che consente lo sviluppo dell'interfaccia grafica di applicazioni desktop utilizzando tecnologie tipicamente pensate per il web, quali *HTML*, *CSS*, *JavaScript* e *TypeScript*; per fare ciò, questa tecnologia combina il motore di rendering *Chromium* e il *runtime NodeJS*.

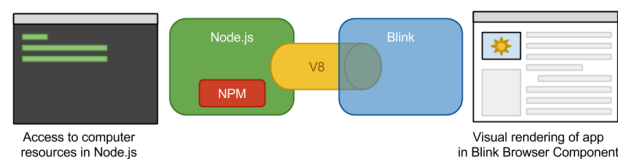


Figura 1.9: Funzionamento del *framework Electron* con *NodeJS*.

Fonte: freecontent.manning.com

Sync Lab utilizza strumenti tecnologici anche per gestire il lavoro da remoto; in particolare, durante l'emergenza sanitaria ancora in corso, l'azienda fa uso di diverse tecnologie per organizzare il lavoro non in presenza, permettendo a tutti i collaboratori di rimanere correttamente aggiornati sulle attività e i compiti di cui sono responsabili. Alcune di queste tecnologie sono le seguenti:

- * **Google Meet:** servizio di *Google* per effettuare videoconferenze online. Questa piattaforma viene usata dall'azienda, e in particolare è stata utilizzata anche durante il mio *stage*, per comunicare con gli altri collaboratori e poter rimanere aggiornati sui progressi effettuati;
- * **Discord:** applicazione *VoIP* per la comunicazione vocale e testuale. Uno dei punti di forza di questa piattaforma, sfruttato anche dall'azienda, è la possibilità di poter avere più canali, sia testuali che vocali, all'interno dello stesso *server*; questo permette una comunicazione più ordinata e metodica, riducendo il rischio di incomprensioni;
- * **Google Docs:** servizio di *Google* per la condivisione di documenti online. Questa piattaforma è stata utilizzata in particolare durante il mio tirocinio per tenere traccia degli incrementi giornalieri che ho svolto;
- * **Trello:** software gestionale in stile *Kanban*, utilizzato per coordinare il proprio *workflow* e visualizzare quello degli altri collaboratori. Questa piattaforma si sposa bene con la metodologia *Agile* che Sync Lab utilizza, in quanto può essere utilizzato come una *Scrum board*.

1.5 Propensione all'innovazione

Sync Lab presta grande attenzione anche all'innovazione e allo sviluppo, sia in senso tecnologico che industriale. L'azienda, infatti, conta tre dipartimenti ideati per sperimentare e innovare, i quali sono **Research and Development**, nato con lo scopo di promuovere nuovi prodotti nati da ricerche in svariati settori, **Lab**, in cui l'azienda sviluppa soluzioni a quanto studiato nel precedente dipartimento, e **Start-up**, il cui scopo è quello di promuovere le *start-up* di maggiore rilevanza per quanto riguarda l'innovazione; per fare ciò, Sync Lab collabora con svariati enti privati e università, sia italiane che estere.



Figura 1.10: Alcune università con le quali l'azienda collabora.

Fonte: synclab.it

Alcuni dei progetti di ricerca fondati e mantenuti da Sync Lab sono i seguenti:

- * **BIG-ASC:** acronimo di *BIG Data and Advanced Analytics for Secure Mobile Commerce*, è un progetto che punta a creare una piattaforma *Big Data* che sappia rispondere a requisiti stringenti delle piattaforme di *Mobile Commerce*, come la scalabilità, l'autonomia e le performance, attraverso l'analisi continua e in tempo reale dei dati d'utilizzo. Per fare questo, Sync Lab collabora con l'azienda *CeRICT* e l'*Università degli studi Parthenope* di Napoli;
- * **eHealthNet:** ecosistema software per la Sanità Elettronica, che si propone di intervenire su quattro aree tematiche riguardanti la sanità, ossia interoperabilità, pervasività, sostenibilità e preventivabilità. Per lo sviluppo di questo progetto è stato avviato un laboratorio che vede come collaboratori svariate aziende private, l'*Istituto Italiano di Tecnologia*, l'*Istituto Nazionale Tumori*, l'*Università degli studi Federico II* di Napoli e l'*Università degli studi di Salerno*;
- * **BDA4PHR:** piattaforma *open-source*, scalabile, estendibile e manutenibile che offre servizi di *storing* e *Big Data Analytics* dedicati ad informazioni di tipo medico-sanitario. Scopo di questo progetto è la creazione di un *repository* sicuro, distribuito e affidabile per la gestione, la condivisione e l'analisi di dati eterogenei. Tra i finanziatori di questo progetto ci sono l'*Unione Europea* e il *Ministero dello Sviluppo Economico* italiano.



Figura 1.11: I progetti di ricerca e sviluppo a cui Sync Lab collabora.

Fonte: synclab.it

Sempre nell'ambito dell'innovazione, posso dire che l'azienda ha un atteggiamento molto propositivo e aperto alle nuove tecnologie: non è raro, infatti, che i collaboratori propongano l'utilizzo di nuovi linguaggi, *framework* e strumenti per completare i servizi e i prodotti offerti dall'azienda.

Ho potuto respirare questo clima di apertura anche nell'ambito del mio tirocinio: avendo avuto accesso alla piattaforma *Discord* aziendale, ho potuto notare che il *server* è diviso in più sottocanali, ognuno dedicato a un ambito di sviluppo, in cui i dipendenti inoltrano articoli e documentazione riguardanti nuove tecnologie, aprendo così a un dibattito costruttivo.

Capitolo 2

Il progetto nel contesto aziendale

2.1 Il rapporto tra stage e azienda

Lo *stage* è un momento fondamentale nella carriera di uno studente universitario; esso infatti rappresenta un passaggio dal mondo accademico al mondo del lavoro, permettendo di passare gradualmente da un'ottica di studio prettamente teorico, per quanto corredato da esercizi pratici, a un'ottica di lavoro con tecnologie, strumenti e metodologie utilizzate quotidianamente dalle aziende.

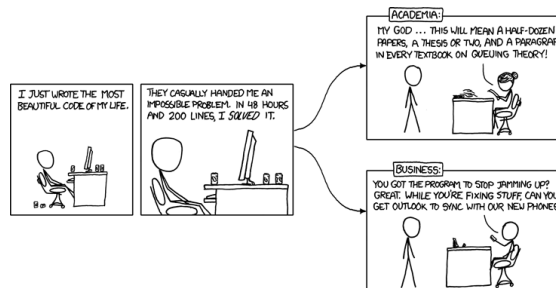


Figura 2.1: *Academia vs business - xkcd.*

Fonte: xkcd.com

Sync Lab, avendo sedi in città universitarie ed essendo sempre tesa verso l'innovazione, prende con gran serietà l'opportunità di ospitare tirocini al suo interno. Questo si può vedere dalla grande **esperienza** che questa azienda ha in fatto di *stage*: essi sono infatti organizzati in modo da conformarsi al meglio alle esigenze dello studente, in quanto molto flessibili in termini di durata complessiva. In Sync Lab, inoltre, lo studente non solo è seguito da un tutor esterno, che si occupa di guidare il tirocinante sia organizzativamente che pragmaticamente durante lo sviluppo, ma può trovare aiuto e consiglio anche negli altri collaboratori dell'ufficio.

Per quanto riguarda le motivazioni che ha Sync Lab ad ospitare *stage* universitari, ho individuato le seguenti:

- * Prima tra tutte è la tendenza all'**innovazione**: uno studente universitario, infatti,

per quanto inesperto della vita lavorativa è mediamente più aggiornato sulle nuove tecnologie, e questo porta l'azienda ad aprirsi a nuove tematiche;

- * C'è poi un'ottica di **assunzione futura**: le università sono infatti per Sync Lab il principale bacino di nuovi lavoratori. È molto frequente, infatti, che dopo un tirocinio ben riuscito l'azienda offra un contratto di lavoro allo studente, se questi è disponibile al lavoro;
- * Infine, un aspetto da prendere in considerazione è quello **economico**: l'azienda, infatti, non è obbligata a pagare un rimborso spese al tirocinante, e la maggior parte delle spese assicurative sono a carico dell'università; la presenza di uno o più tirocinanti, infine, non rappresenta un costo rilevante in termini di risorse.

2.2 L'azienda in relazione al contesto attuale

Il 31 dicembre 2019, le autorità cinesi riferiscono all'*Organizzazione Mondiale della Sanità* di 41 casi di polmonite anomala che si sono verificati a Wuhan, capitale della provincia di Hubei, in Cina; pochi giorni dopo, gli scienziati cinesi identificano il nuovo virus con il nome di *2019-nCov*. Inizialmente, tutto il mondo sottovaluta questa situazione, pensando che fosse un problema riguardante solo la Cina e pochi altri paesi del sud-est asiatico; il 31 gennaio 2020, però, vengono confermati i primi due casi di contagio in Italia.¹ Da quel momento, la sensazione che potesse diventare un problema globale comincia a farsi strada nei pensieri di molte persone.

I giorni a seguire vedono una rapida successione di eventi: a inizio febbraio il virus viene rinominato dall'OMS in *SARS-CoV-2*, a fine febbraio scattano le cosiddette **zone rosse** in Italia, zone da cui diventa impossibile uscire e in cui è impossibile entrare, il 7 marzo la Lombardia diventa totalmente zona rossa e, due giorni dopo, scatta il *lockdown* in tutta Italia². Da questo momento, la maggior parte dei luoghi di lavoro viene chiusa, e questo *shutdown* dura fino all'introduzione della *fase 2*, il 4 maggio, giorno in cui alcune attività vengono riavviate, e ancora parzialmente fino al 15 giugno, giorno in cui scatta la *fase 3* che durerà fino al 13 ottobre.

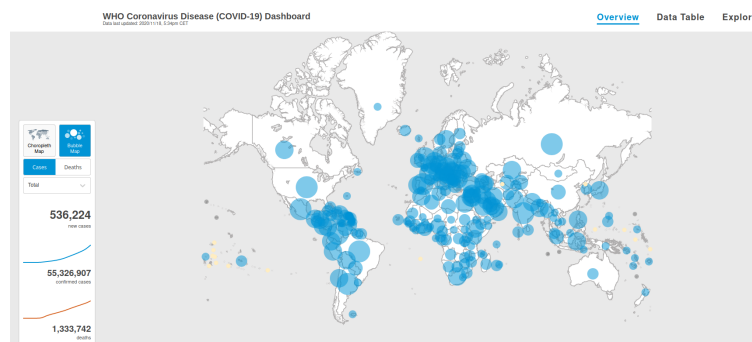


Figura 2.2: Dashboard di COVID-19 nel mondo.

Fonte: covid19.who.int

¹Il Sole 24 Ore. *Cronistoria del Coronavirus*. URL: <https://lab24.ilsole24ore.com/storia-coronavirus/>.

²Ibid.

Durante questo periodo di emergenza sanitaria, molte aziende cominciano ad adottare le tecniche del *remote working* e dello *smart working*; tra queste aziende c'è anche Sync Lab, che si attrezza per permettere ai suoi dipendenti e ai tirocinanti di lavorare da casa, interrompendo inizialmente e riducendo di molto successivamente la presenza fisica negli uffici.

Questo contesto di attualità è di grande importanza anche per quanto riguarda il mio *stage* in particolare: temporalmente parlando, infatti, il mio tirocinio è coinciso con la cosiddetta fase 3. Sebbene fosse possibile effettuare incontri fisici, l'azienda ha infatti deciso di ridurre tali incontri per una questione di sicurezza; da questo ne deriva che ho svolto gran parte del mio tirocinio da casa, senza poter accedere direttamente alle strutture e alle infrastrutture se non una volta alla settimana.

Oltre a questo, il periodo è altresì di vitale importanza per descrivere lo **scopo** del mio tirocinio: come descrivo più approfonditamente nella sezione seguente, infatti, il progetto di *stage* si configura all'interno del progetto *SyncTrace*, un insieme di applicativi concernenti il tema del *contact tracing*.

2.3 Lo scopo dello stage

In risposta alla situazione sanitaria, molte aziende del settore *ICT* si sono attivate per sviluppare applicazioni e sistemi di *contact tracing*. Il *contact tracing* consiste nell'identificazione delle persone che potrebbero essere venute a contatto con una persona infetta, e nella successiva raccolta di ulteriori informazioni, quali vicinanza e tempo di esposizione, su tali contatti; il fine di questa pratica è aiutare il sistema sanitario a individuare i possibili contagi, arrivando a costruire un albero dei contatti per poter agire tempestivamente sul problema.



Figura 2.3: Rappresentazione schematica del *contact tracing*.

Fonte: mashable.com

Tra le aziende che si sono mobilitate per questa causa c'è anche Sync Lab, che ha avviato un suo progetto in tale ambito chiamato *SyncTrace*. Questo sistema ha come scopo lo sviluppo di due prodotti differenti ma in relazione tra loro: un'applicazione di *contact tracing* puro e un applicativo di gestione e organizzazione per i medici e per gli esercenti di attività commerciali.

Il **primo** applicativo, pensato per l'utilizzo da parte di tutta la popolazione, ha come unico scopo il tracciamento dei contatti delle persone infette; esso è in forma di applicazione per dispositivi mobili, quali *smartphone* e *tablet*, ed è pensato in modo da

tracciare i contatti e i livelli di rischio di positività basandosi su parametri oggettivi come la vicinanza e il tempo di esposizione.

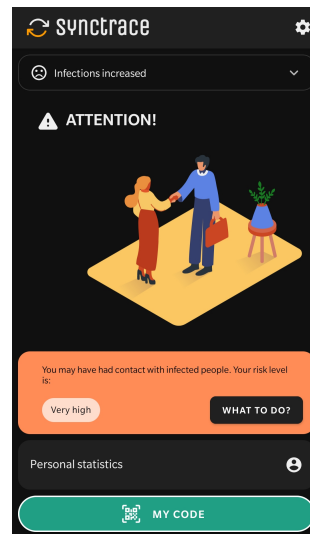


Figura 2.4: Schermata iniziale dell'applicazione *SyncTrace*.

Il **secondo** applicativo, consistente in una *web application*, è pensato per fornire un'interfaccia di gestione del sistema ai medici e come applicazione di controllo per gli esercenti. Esso ha quindi un doppio utilizzo:

- * I **medici**, una volta identificati come tali, possono usarlo per registrare nuovi contatti infetti, eliminare eventuali pazienti non infetti e verificarne la presenza e/o il livello di rischio di contagio nel sistema condiviso;
- * Gli **esercenti**, una volta registrati come tali, possono controllare l'eventuale positività o il livello di rischio di contagio dei clienti che entrano nelle proprie attività.

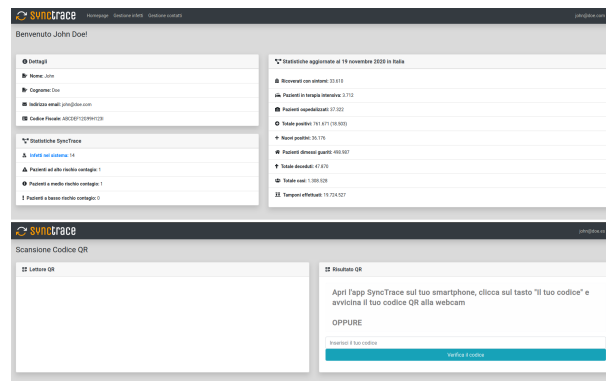


Figura 2.5: Schermata iniziale della *webapp* per, rispettivamente, medici ed esercenti.

Il mio progetto di *stage* si inserisce nel contesto di quest'ultimo applicativo descritto. Lo scopo principale del mio tirocinio è, infatti, effettuare il *porting* dell'applicativo utilizzando il *framework Ionic*, per poter riutilizzare parte della logica già sviluppata secondo il principio del *write once, run anywhere*.

2.4 Vincoli e obiettivi dello stage

Prima di cominciare il tirocinio ho definito, insieme al tutor aziendale Andrea Giunta, gli obiettivi e i vincoli che il mio lavoro avrebbe dovuto rispettare. Questo ha portato alla stesura di un **piano di lavoro**, riportante lo scopo dello stage, i prodotti attesi, la pianificazione settimanale del lavoro e gli obiettivi del progetto; durante il progetto, però, questo piano di lavoro ha subito due modifiche per meglio adattarsi a esigenze personali e per occupare al meglio il tempo in termini di produttività.

2.4.1 Vincoli temporali

Inizialmente, lo *stage* aveva una durata prevista di 8 settimane, di cui 7 da 40 ore l'una e l'ultima da 20 ore, per un totale di 300 ore. A causa di malattia personale, però, non ho potuto effettuare la penultima settimana come previsto, e ho dovuto quindi effettuare una nona settimana per concludere le 300 ore previste e, con esse, quanto previsto dal piano di lavoro. Il totale di ore è stato quindi **308**.

Come già detto, la maggior parte del progetto si è svolto in *smart working*: ho quindi lavorato una media di 8 ore giornaliere da casa. I giorni in cui mi sono recato in ufficio, generalmente il venerdì di ogni settimana, l'orario di lavoro è stato dalle 09.00 alle 18.00, con un'ora di pausa pranzo. Di seguito viene riportata una tabella riassuntiva delle attività svolte, divise per ore e settimane.

Settimana	Ore	Attività
1	8	Formazione sul linguaggio <i>Java SE</i>
	16	Formazione sul linguaggio <i>Java EE</i>
	16	Formazione sul <i>framework Spring</i>
2	40	Formazione sul <i>framework Spring</i>
3	24	Formazione sul linguaggio <i>JavaScript</i>
	8	Formazione sul linguaggio <i>TypeScript</i>
	8	Formazione sul <i>runtime NodeJS</i>
4	40	Formazione sul <i>framework Angular</i>
5	20	Formazione sul <i>framework Ionic</i> e su quanto fatto nel progetto precedente
	4	Configurazione dell'ambiente di produzione
	16	Codifica delle maschere previste dal progetto
6	24	Codifica delle maschere previste dal progetto
	16	Configurazione del server di <i>back-end</i> tramite containerizzazione con piattaforma <i>Docker</i>
7	8	Configurazione del server di <i>back-end</i> tramite containerizzazione con piattaforma <i>Docker</i>
8	16	Codifica delle ultime maschere e di componenti aggiuntive
	24	Verifica del codice scritto
9	8	Collaudo dell'applicativo prodotto
	24	Documentazione dell'intero progetto
	8	Codifica e verifica di ultimi <i>fix</i> a <i>bug</i> trovati tramite il collaudo

Tabella 2.1: Suddivisione settimanale delle attività

2.4.2 Vincoli organizzativi

Prima di cominciare lo *stage* ho inoltre pattuito i vincoli organizzativi con il tutor aziendale. Come già detto, per quanto riguarda questo frangente è stato deciso di lavorare in *smart working* per la maggior parte della durata del progetto; è stato inoltre deciso di incontrarci periodicamente, solitamente il venerdì, per fare il punto della situazione e allinearsi con il tutor aziendale.

Il lavoro che ho effettuato, sia da remoto che in presenza, è stato rendicontato in un foglio di calcolo online, riportante la data, le attività svolte in tale data e l'approvazione o meno da parte del tutor aziendale; ho aggiornato questo documento quotidianamente, e settimanalmente ho ricevuto il *feedback* da parte del tutor.

	A	B	C
1	Data	Argomenti	Verificato
2	31/08/2020	Ripasso del linguaggio Java SE: sintassi, costrutti, buone pratiche di programmazione	OK
3	01/09/2020	Studio pratico del linguaggio Java SE [traduzione di un programma esistente da C++ a Java per esercizio], breve overview di Java EE	OK
4	02/09/2020	Studio teorico di Java EE (videolezioni) + primo approccio pratico allo sviluppo di servlet con Java EE e glassfish	OK
5	03/09/2020	Conclusione studio generale di Java EE con annessi piccoli esercizi pratici riguardanti JPA. Inizio studio teorico del framework Spring e configurazione del sistema per i futuri esercizi pratici su questo [installazione e configurazione di ide, framework, database mysql, server tomcat], e per il futuro sviluppo dell'applicativo richiesto da piano di lavoro	OK
6	04/09/2020	Fine della configurazione del sistema per i futuri esercizi (risolte problematiche con mysql), inizio dello studio dello spring framework (panoramica dell'architettura e del funzionamento generale, creazione del primo progetto di test)	OK
7	07/09/2020	Inizio del secondo progetto di test (da cui si espande poi tutto il resto): introduzione allo spring mvc, creazione di un template con apache tiles e inizio dello studio dello strato di persistenza (connessione dell'applicativo di prova con il database)	OK
8	08/09/2020	Continuazione dello studio dello spring framework, in particolare creazione dello strato di persistenza e dello strato di servizio dell'applicativo, con precedente configurazione della fonte dati (database mysql). Creazione delle prime query, con anche utilizzo delle stored procedures, e inizio delle prime view	OK
9	09/09/2020	Continuazione dello studio dello spring framework, in particolare continuazione della vista con dati prelevati dal database e form di inserimento dati; studio delle diverse notazioni del controller nello spring framework; panoramica sulla sicurezza (prevalentemente prevenzione di sql injection)	ok
10	10/09/2020	Continuazione dello studio dello spring framework, in particolare internazionalizzazione, perfezionamento della vista, JPA2	ok
11	11/09/2020	Studio di spring Boot e dei web services; studio del formato json e suo utilizzo nell'ambito di architetture rest con Spring Boot; full stack development con il framework spring (e spring boot) e angularjs. Conclusione dello studio di Spring	ok

Figura 2.6: Porzione del registro delle attività.

Per quanto riguarda il salvataggio e il versionamento del codice e della documentazione prodotti, inoltre, ho fatto uso della *repository GitLab* aziendale.

2.4.3 Vincoli tecnologici

Per quanto riguarda i vincoli tecnologici, posso dire di essere stato abbastanza libero nella scelta di quali strumenti utilizzare; i vincoli a cui sono sottostato sono quindi stati decisi principalmente da me, in accordo con il tutor aziendale. Questi sono qui riportati.

Sistema operativo

Essendo il prodotto a cui ho lavorato non coperto da segreto industriale, ho svolto l'intero progetto sulla mia macchina senza bisogno di software esterni per rispettare le norme di sicurezza aziendali. Il sistema operativo che ho utilizzato per lo sviluppo è stato *GNU/Linux*, in particolare la distribuzione *Arch Linux* a 64 bit.

PostgreSQL

Sebbene il mio lavoro non comprendesse l'utilizzo diretto di un *database*, la componente *back-end* dell'applicativo fa uso del *database* relazionale *PostgreSQL*; ho avuto modo

di utilizzare questa tecnologia durante il processo di collaudo, per verificare che i dati inseriti dall'applicazione venissero correttamente immagazzinati dal *database*.

Docker

Per effettuare il *deploy* della componente *back-end* dell'applicativo, necessaria al processo di collaudo su dispositivo mobile, ho fatto uso dello strumento di containerizzazione *Docker*. Questo mi ha permesso di rendere la suddetta componente installabile facilmente su qualsiasi sistema.

Android Studio

Per effettuare il collaudo di quanto sviluppato su dispositivo mobile, sono stato vincolato all'ambiente di sviluppo integrato *Android Studio*, fornito gratuitamente dall'azienda *JetBrains*, per la compilazione dell'applicazione per dispositivi *Android*; in accoppiata a questo ho utilizzato il tool *ADB* (*Android Debug Bridge*) per l'installazione dell'applicazione sul dispositivo.

Documentazione

Per quanto riguarda la documentazione, dopo i consigli del mio tutor aziendale la mia scelta è ricaduta sul tool *Compodoc*, uno strumento simile a *Javadoc* per applicazioni sviluppate tramite il framework *Angular*; questo strumento ha permesso la generazione automatica di documentazione a partire da commenti strutturati al codice.

2.4.4 Obiettivi

Insieme al tutor aziendale ho delineato gli obiettivi dello *stage* e li ho riportati in un piano di lavoro validato dal tutor interno, il prof. Tullio Vardanega. Ogni obiettivo possiede un nome formato nel seguente modo:

0-[tipologia] [numero]

dove:

- * [tipologia] indica la tipologia di obiettivo. Questa può essere:
 - 0: obbligatorio, ossia un obiettivo vincolante e necessario da soddisfare;
 - D: desiderabile, ossia un obiettivo non vincolante ma dal riconoscibile valore aggiunto;
 - F: facoltativo, ossia un obiettivo non vincolante e rappresentante un valore aggiunto non competitivo.
- * [numero] consiste in un numero, progressivo per ogni tipologia.

Gli obiettivi inizialmente individuati sono riportati nella seguente tabella.

Obiettivo	Descrizione
O-01	Acquisizione delle competenze su linguaggi di programmazione, frameworks e strumenti di supporto
O-02	Capacità di raggiungere gli obiettivi richiesti in autonomia seguendo il cronoprogramma
O-03	Portare a termine le implementazioni previste con una percentuale di superamento pari all'80%
O-D1	Portare a termine le implementazioni previste con una percentuale di superamento pari al 100%
O-F1	Dare un contributo importante nelle fasi di progettazione delle maschere con l'obiettivo di realizzare una app usabile, responsive e adaptive
O-F2	Implementare la lettura del QR Code direttamente dalla fotocamera dello smartphone, evitando quindi inserimenti manuali

Tabella 2.2: Obiettivi di progetto originali.

In corso di progetto il piano di lavoro è stato rimodulato, in quanto ho svolto parte del lavoro in un tempo minore del previsto; ai precedenti obiettivi si è quindi aggiunto il seguente:

Obiettivo	Descrizione
O-D2	Containerizzazione della componente <i>back-end</i> del progetto <i>SyncTrace</i>

Tabella 2.3: Ulteriore obiettivo, aggiunto in corso di progetto.

Il piano di lavoro identifica inoltre i prodotti attesi; questi sono:

- * Un **documento tecnico** che descriva le maschere utilizzate;
- * Il **codice** rilasciato sul *repository* indicato dall'azienda.

2.5 Motivazione della scelta

Sono venuto a conoscenza di Sync Lab durante l'evento **StageIT 2020**. Questo evento, organizzato dal professor Tullio Vardanega dell'*Università degli Studi di Padova* in collaborazione con l'associazione di imprenditori *Assindustria Venetocentro*, è un'occasione in cui gli studenti di diverse facoltà dell'università di Padova vengono messi in contatto con le aziende presenti sul territorio.

L'evento viene organizzato annualmente; quest'anno, però, a causa dell'emergenza sanitaria non si è potuto svolgere in presenza. Nonostante questo, l'adesione da parte delle aziende e degli studenti è stato comunque soddisfacente, anche grazie al fatto che, seguendo gli incontri con le aziende per via telematica, è stato possibile contattarne più di una contemporaneamente.



Figura 2.7: Logo di StageIT 2020.

Fonte: assindustriavenetocentro.it

Durante l'evento *StageIT* ho avuto occasione di parlare con i rappresentanti di quattro aziende operanti nel comune di Padova, tra cui l'ing. Fabio Pallaro di Sync Lab.

Gli ambiti di *stage* a cui ero interessato erano principalmente tre: sviluppo software *mobile*, *data science* e *cybersecurity*. Uno dei motivi per cui ho optato per Sync Lab come azienda è stato che l'ing. Pallaro mi ha offerto due diversi ambiti di *stage*: uno riguardante lo sviluppo di un'applicazione per dispositivi mobili tramite *framework Ionic*, e uno riguardante l'aspetto di sicurezza dei prodotti che l'azienda stava e avrebbe sviluppato con altri tirocinanti, entrambi ambiti a cui ero particolarmente interessato. L'azienda, inoltre, mi è sembrata molto attenta agli studenti interessati a un tirocinio presso di essa, segno a mia opinione di serietà, professionalità e apertura a nuove esperienze.

Per quanto riguarda il progetto, ho infine optato per lo sviluppo di un applicativo mobile per svariati motivi, anzitutto l'ambito: la tematica del *contact tracing* mi è infatti fin da subito interessata per l'utilità che può avere in situazioni come quella attuale. Oltre a questo, ho trovato che le tecnologie che sarei poi andato ad utilizzare fossero interessanti da un punto di vista personale e, soprattutto, professionale: i linguaggi *JavaScript* e *TypeScript* e i *framework* web e *mobile* come *Angular* e *Ionic*, infatti, sono utilizzati sempre di più frequentemente dalle aziende operanti nel settore *IT*.

2.6 Formazione

Ho dedicato il primo mese del mio tirocinio allo studio e all'approfondimento di tutte le conoscenze che mi sarebbero servite, o mi sarebbero state utili, a sviluppare il progetto di *stage* vero e proprio. Gli **strumenti** che ho utilizzato a tale scopo sono molteplici: anzitutto, ho fatto uso della piattaforma *Udemy* per acquisire le informazioni teoriche riguardanti le tecnologie oggetto di studio; a questo ho associato anche l'utilizzo della piattaforma *YouTube*, sulla quale ho trovato svariati video di particolare utilità. Per quanto riguarda gli esercizi pratici, invece, ho fatto uso sia della piattaforma *Udemy* che di *tool* per l'esercitazione, quali ad esempio il pacchetto *npm "LearnYouNode"* per quanto riguarda lo studio del *runtime NodeJS*.

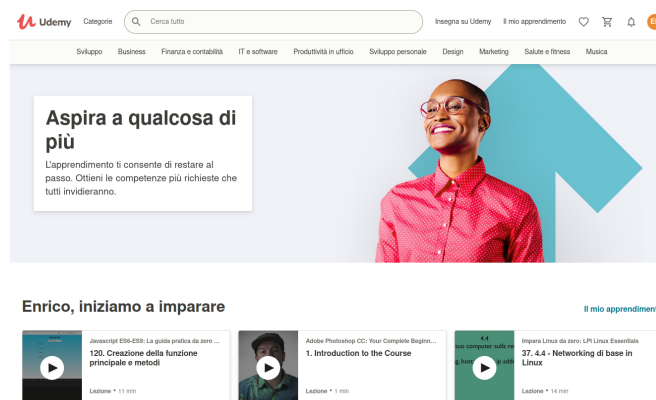


Figura 2.8: Homepage della piattaforma *Udemy*.

Fonte: [udemy.com](https://www.udemy.com)

2.6.1 Tecnologie

In questa sezione sono elencate le tecnologie che ho studiato e appreso; in particolare, ad ogni tecnologia elencata è associata la descrizione del suo utilizzo nell'ambito dello stage.

Per quanto riguarda i **linguaggi di programmazione**, quelli che ho approfondito sono i seguenti:

- * **Java**: di questo linguaggio ho studiato principalmente la sintassi, i costrutti di base e, soprattutto, l'implementazione di *servlet* e le *JPA* (*Java Persistence API*) del linguaggio *Java Enterprise Edition*. Sebbene il mio progetto non prevedesse lo sviluppo in codice *Java*, la componente *back-end* del prodotto cui il mio progetto appartiene è scritto con questo linguaggio in accoppiata al *framework Spring*; ho quindi ritenuto opportuno, insieme al tutor aziendale, approfondire questo argomento per poter comprendere almeno a grandi linee il funzionamento del *back-end*;
- * **JavaScript**: prima di cominciare lo *stage* avevo già delle conoscenze abbastanza radicate di questo linguaggio; di questo ho quindi studiato aspetti avanzati, quali *promises*, *fetch api* e le funzioni asincrone. Sebbene anche questo linguaggio non fosse previsto dal mio progetto, è stato altresì utile da studiare come base per lo studio di *TypeScript*;
- * **TypeScript**: questo linguaggio è stato quello principalmente utilizzato per lo sviluppo dell'applicativo del mio progetto. Avendo già studiato *JavaScript*, di cui *TypeScript* è un *super-set*, lo studio è stato semplice e veloce.

Lo studio di ogni linguaggio di programmazione è stato inoltre associato allo studio di uno o più **framework**; questi sono:

- * **Spring**: di questo *framework*, utilizzato in ambito *Java*, ho studiato a grandi linee il funzionamento generale, comprendente i diversi strati da cui è composto un applicativo sviluppato con questa tecnologia (*database*, persistenza, *business* e presentazione). Come riportato precedentemente, il mio progetto non prevedeva lo sviluppo in codice *Java* ma è stato utile per comprendere il funzionamento della componente *back-end*, e lo stesso vale per il *framework Spring*;
- * **Angular**: questo *framework*, utilizzato con linguaggio *TypeScript*, è la tecnologia utilizzata per lo sviluppo dell'applicativo di cui il mio progetto prevedeva il *porting*; è stato quindi necessario studiarlo per capire al meglio come fosse strutturato tale applicativo. Oltre a questo, il *framework Ionic*, con il quale ho sviluppato il mio progetto, prevede l'utilizzo di un sottostrato *Angular*; ho quindi usato questo *framework* anche per lo sviluppo in sé;
- * **Ionic**: questo *framework*, pensato per lo sviluppo di *PWA* (*Progressive Web Apps*) con codice *TypeScript*, è stato quello più da me utilizzato per lo sviluppo. Ho studiato questo linguaggio tramite la documentazione fornita dall'*Ionic Team*, in quanto a mio parere esaustiva e ben strutturata.

Per completare le mie conoscenze in ambito *back-end JavaScript*, inoltre, ho studiato il runtime **NodeJS**. Nessun componente del prodotto a cui il mio progetto appartiene è sviluppato interamente con *NodeJS*; alcune componenti di esso vengono utilizzate, però, per la gestione del collegamento tra la componente *front-end* e quella *back-end*

dell'applicativo di cui ho eseguito il *porting*.

Ho infine approfondito **Docker**, piattaforma che automatizza il *deployment* di applicazioni all'interno di macchine virtuali *linux* minimali. Ho approfondito questo argomento a sviluppo già cominciato, poiché per il processo di collaudo si è reso necessario per me avere la componente *back-end* del prodotto su un server accessibile via *http*, e mi è stato proposto di containerizzare tale componente per poter effettuare un *deployment* più efficiente.

2.6.2 Progetto

Come precedentemente riportato, il mio progetto è consistito nell'effettuare il *porting* di una *web application* su dispositivi mobili tramite il *framework Ionic*. L'applicazione web, consistente in una *Single Page Application* sviluppata in *TypeScript* tramite il *framework Angular*, offre le stesse funzionalità dell'applicativo sviluppato da me; molte di queste, però, non sono traslabili direttamente su *mobile*, poiché consistenti di elementi non nativi dei sistemi operativi mobili. Esempio di ciò sono sia piccolezze estetiche, quali ad esempio i bottoni, i *form* e gli *alert*, che l'utilizzo delle risorse hardware del sistema, quali ad esempio la fotocamera, i bottoni fisici e gli strumenti di misura come l'accelerometro. Ho quindi effettuato anche uno studio su quanti e quali componenti software avrei dovuto cambiare per sviluppare correttamente il mio prodotto.

Capitolo 3

Il progetto di stage

3.1 Analisi dei requisiti

Sezione contenente l'analisi dei requisiti del prodotto software. In questa sezione verranno riportati anche i diagrammi dei casi d'uso dell'applicativo e, possibilmente, i *mockup* dell'applicazione creati prima di cominciare le attività di progettazione e codifica.

3.2 Progettazione

Sezione contenente le scelte progettuali che ho preso, insieme al tutor esterno, per lo sviluppo dell'applicativo. Poiché gran parte delle scelte è stata presa durante il tirocinio antecedente al mio, questa sezione non avrà una grande estensione; nonostante questo, alcune scelte progettuali sono state prese da me, e ritengo quindi giusto riportarle.

3.3 Codifica

Sezione contenente una descrizione dell'attività di codifica applicata al mio progetto.

3.4 Verifica

Breve introduzione all'attività di verifica svolta, in cui riassumerò le modalità adottate nel caso del mio progetto software e gli strumenti utilizzati per il *testing*.

3.4.1 Analisi statica

Descrizione dell'utilizzo dell'analisi statica per il controllo di conformità del codice agli standard. In questa sottosezione parlerò anche degli strumenti utilizzati per effettuare l'analisi statica del codice e dei risultati ottenuti in quest'ambito.

3.4.2 Test di unità

Sezione riportante tutte le informazioni riguardanti i test di unità implementati e i risultati delle misure.

3.5 Validazione e collaudo

Sezione contenente l'analisi delle attività di validazione eseguite. La sezione sarà divisa in due sottosezioni non numerate: la prima riguarderà la validazione, e quindi conterrà l'analisi dei requisiti soddisfatti; la seconda conterrà l'analisi del collaudo. In quest'ultima sottosezione parlerò anche della containerizzazione del back-end di SyncTrace, poiché è stata un'attività fondamentale per poter eseguire il collaudo dell'applicazione su un dispositivo mobile.

3.6 Risultati ottenuti

Sezione contenente una visione d'insieme dei risultati ottenuti, ossia un riassunto di quanto sviluppato, verificato, validato e documentato.

Capitolo 4

Valutazioni retrospettive

4.1 Soddisfacimento degli obiettivi

Elenco formale degli obiettivi soddisfatti e descrizione del grado di soddisfacimento di questi. In questo capitolo riassumerò anche i prodotti di stage completati, ossia applicazione, containerizzazione del *back-end* e documentazione.

Il bilancio degli obiettivi verrà trattato sia dal punto di vista degli obiettivi pattuiti con l'azienda che dal punto di vista dei miei obiettivi personali.

4.2 Bilancio formativo

4.2.1 Maturazione professionale

Descrizione delle competenze professionali acquisite, sia in termini di tecnologie imparate che di abilità nella gestione del lavoro. In questo capitolo parlerò dunque dei linguaggi di programmazione, dei framework e delle infrastrutture che ho conosciuto e imparato a utilizzare e/o di cui ho imparato l'utilizzo in un contesto reale come è il tirocinio in azienda.

4.2.2 Rapporto tra università e lavoro

Trattazione delle differenze tra quanto appreso in ambito accademico e quanto utilizzato durante il tirocinio in azienda.

Nella stesura di questo capitolo utilizzerò un doppio approccio alla trattazione: da un lato parlerò delle tecnologie utilizzate che non vengono insegnate durante la laurea triennale; dall'altro parlerò della distanza che ho riscontrato tra le conoscenze "puramente accademiche" e l'applicazione di queste in contesto di lavoro reale.