

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



Chatbot di Facebook Messenger per la risposta a domande frequenti

Tesi di laurea triennale

Relatore

Prof. Paolo Baldan

Laureando

Mauro Carlin

ANNO ACCADEMICO 2016-2017

I'm personally convinced that computer science has a lot in common with physics. Both are about how the world works at a rather fundamental level. The difference, of course, is that while in physics you're supposed to figure out how the world is made up, in computer science you create the world. Within the confines of the computer, you're the creator. You get to ultimately control everything that happens. If you're good enough, you can be God. On a small scale.

— Linus Torvalds

Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage dal laureando Mauro Carlin presso l'azienda i-contact S.r.l di Belluno (BL). Lo stage è stato svolto alla conclusione del percorso di studi della Laurea Triennale ed è durato in totale 300 ore.

Gli obiettivi da raggiungere erano molteplici. In primo luogo l'azienda ha richiesto un'analisi dei principali [NLP](#) presenti sul mercato, in modo da valutarne pregi e difetti. Questo strumento viene utilizzato per trasformare le domande di un utente in dati processabili.

Il passo successivo è stato studiare ed integrare questo sistema in due chatbot di Facebook Messenger creati e gestiti dall'azienda stessa, per dare la possibilità all'utente di interagire con essi anche tramite domande di senso compiuto, e non solo tramite le possibilità offerte dalla piattaforma Facebook.

Il presente documento descrive nei primi due capitoli il contesto aziendale e come questo progetto di stage possa essere stato utile per l'azienda. Il terzo capitolo documenta lo svolgimento dello stage descrivendo le attività che sono state portate a termine, i punti salienti del progetto stesso e le principali scelte progettuali. Il quarto ed ultimo capitolo presenta infine una valutazione dello svolgimento dello stage rispetto agli obiettivi aziendali e alle conoscenze acquisite dallo studente.

Ringraziamenti

Innanzitutto, vorrei esprimere la mia gratitudine al Prof. Paolo Baldan, relatore della mia tesi, per l'aiuto e il sostegno fornitomi durante la stesura del lavoro.

Desidero ringraziare con affetto i miei genitori Rita e Roberto per il sostegno, il grande aiuto e per essermi stati vicini in ogni momento durante gli anni di studio.

Ringrazio tutti gli amici conosciuti durante questo percorso, in particolare Luca, Pier Paolo, Nicola, Mattia, Marco, Simeone e Tommaso, per aver reso questi tre anni indimenticabili.

Infine ringrazio tutti coloro che mi hanno aiutato, anche con un piccolo gesto, a raggiungere questo meraviglioso traguardo.

Padova, Settembre 2017

Mauro Carlin

Indice

1	Il contesto aziendale	1
1.1	Profilo aziendale	1
1.2	Dominio applicativo	1
1.3	Tecnologie utilizzate	1
1.3.1	Applicazioni <i>iOS</i> e <i>Android</i> native	2
1.3.2	Applicazioni web con Java	2
1.3.3	Frontend	2
1.4	Processi aziendali	2
1.4.1	Metodologia	2
1.4.2	Strumenti a supporto dei processi	3
1.5	Clienti	4
1.6	Struttura del documento	5
2	Progetto di stage	7
2.1	Descrizione del progetto	7
2.2	Principali problematiche	9
2.2.1	Difficoltà di ambientazione	9
2.2.2	Difficoltà lavorative	9
2.3	Strumenti utilizzati	10
2.3.1	NetBeans	10
2.3.2	Api.ai	10
2.3.3	Hibernate	11
2.3.4	Database MySQL	11
2.4	Prodotto ottenuto	12
	Glossario	13

Elenco delle figure

2.1	Modello generico di carosello per Facebook Messenger	8
2.2	Logo di NetBeans	10
2.3	Logo di api.ai	11
2.4	Logo di Hibernate	12

Elenco delle tabelle

Capitolo 1

Il contesto aziendale

1.1 Profilo aziendale

i-contact s.r.l. è un'azienda che nasce nel 2003 a Belluno, dove tuttora mantiene la propria sede. Nei primi anni si dedica allo sviluppo di applicativo web per conto di terzi, lavorando in stretta collaborazione con aziende di comunicazione per consulenze di tipo tecnico. Parallelamente inizia la creazione del prodotto di punta dell'azienda: **SMSHosting**¹. Si tratta di un *gateway* per l'invio e la ricezione di sms professionali da web, che consente agli utenti di comunicare con i propri clienti direttamente dalla propria area riservata, oppure dall'esterno tramite email , **FTP**, moduli web, software Windows, app per *smartphone* e molto altro.

Nel corso degli anni, e con l'introduzione di nuove tecnologie, l'azienda evolve i propri prodotti, introducendo nuovi strumenti di comunicazione per i propri clienti, focalizzando le proprie competenze sui mobile che dal 2009 iniziavano a diffondersi. Viene così creata una piattaforma dedicata all'invio di **messaggi push** tramite le più diffuse *messaging apps*, come Facebook Messenger e Telegram. I clienti possono utilizzare questo servizio, integrato nella piattaforma SMSHosting, attraverso email o delle semplici **API REST**.

Come ultima novità i-contact s.r.l. ha iniziato lo sviluppo di **Chatbot** per Facebook Messenger e Telegram. L'azienda mette a disposizione sia una piattaforma semplice e intuitiva dove un utente può creare il proprio **Chatbot** in pochi passi, seguendo i più comuni template di business, sia la possibilità di crearne di nuovi secondo le richieste del cliente.

1.2 Dominio applicativo

1.3 Tecnologie utilizzate

Le tecnologie principali che vengono utilizzate da i-contact s.r.l. per lo sviluppo dei propri prodotti possono essere divise in tre diverse aree:

- * applicazioni *iOS* e *Android* native;
- * applicazioni web con tecnologie *Java*;

¹ *SMSHosting.it*. URL: <https://www.smshosting.it/>.

* *frontend*.

1.3.1 Applicazioni *iOS* e *Android* native

Le tecnologie utilizzate per lo sviluppo di applicazioni mobile dipende naturalmente dal sistema operativo dove si vuole sviluppare. L'azienda utilizza anche dei *framework cross-platform* come *PhoneGap*.

Android

Per quanto riguarda *Android* l'azienda si affida al linguaggio nativo di questo sistema operativo, cioè *Java*. Il team di sviluppo ha grande conoscenza di questo linguaggio, agevolando così lo sviluppo dei nuovi prodotti.

iOS

Per lo sviluppo di applicazioni *iOS* il linguaggio che viene principalmente utilizzato è *Objective-C*, un linguaggio ben noto a coloro che devono codificare questa versione dei prodotti.

1.3.2 Applicazioni web con Java

Le applicazioni web su piattaforma *Java* sono nel DNA di i-contact s.r.l. fin dalla nascita nel 2003.

Il team di sviluppo ha grande conoscenza dei principali *framework* di sviluppo ed ha lavorato a progetti di grande dimensione utilizzando sia CMS open source (Open CMS) che commerciali (Broadvision, Vignette OpenText). Il *framework Spring, Hibernate ORM, Quartz Scheduler, Jersey for RESTful Web services* sono solo pochi esempi delle librerie comunemente utilizzate e che fanno parte del *core* dei loro prodotti.

1.3.3 Frontend

Nello sviluppo della parte *frontend* dei propri portali e applicativi web, i-contact s.r.l. mira ad utilizzare gli strumenti più innovativi per garantire la massima velocità di presentazione e la compatibilità con i *device* di nuova generazione.

I principali linguaggi e *framework* sono *HTML5, CSS3, JQuery, Bootstrap e Sencha*.

1.4 Processi aziendali

1.4.1 Metodologia

Fino dai suoi inizi i-contact s.r.l. crea delle soluzioni software altamente dipendenti dalle specifiche dei propri clienti. Per potere fare ciò è indispensabile mantenere una stretta comunicazione con il cliente, per capire le sue volontà e le sue richieste in modo preciso. La realizzazione di nuovi progetti deve quindi essere in grado di reagire al cambiamento dei requisiti anche in fase di sviluppo, in quanto il cliente può cambiare idee e giudizi sulle funzionalità del proprio prodotto.

La metodologia di sviluppo più adatta, adottata quasi completamente dall'azienda, è la [Adaptive Software Development](#). Il metodo agile utilizzato da i-contact s.r.l. prevede una forte e frequente collaborazione con il cliente, in modo da ricevere dei *feedback* puntuali sugli incrementi portati al prodotto. In ogni momento il cliente si trova ad

avere un prodotto via via più completo, che rispecchia i requisiti da esso imposti e discussi con l'azienda.

La chiave per il successo di questa metodologia è racchiusa in questi punti:

- * sviluppare qualcosa di utile;
- * coltivare la fiducia degli *stakeholders*;
- * costituire gruppi di lavoro competenti e collaborativi;
- * far sì che il team abbia la possibilità e sia in grado di prendere decisioni;
- * consegnare spesso nuove versioni all'aggiunta di nuove funzionalità;
- * incoraggiare l'adattabilità;
- * cercare di ottenere l'eccellenza tecnica;
- * quando possibile, aumentare il volume di dati immessi.

1.4.2 Strumenti a supporto dei processi

Gestione di progetto

Per la **gestione di progetto** i-contact s.r.l. utilizza Asana² per l'assegnazione di *task* relativi a nuove attività da svolgere o *bug* da correggere e BeeBole³ come *web timesheet*.

Asana Asana è un [SaaS](#) *web based* che mira a migliorare la collaborazione all'interno dei team di lavoro. Permette infatti la gestione di progetti e *tasks* online, senza dover utilizzare le email.

i-contact s.r.l. ha creato un proprio *workspace*, dove poter aggiungere nuovi progetti e nuovi *tasks* assegnati a questi progetti.

Per ogni *task* che si vuole creare, è possibile specificare una serie di dettagli:

- * **nome e descrizione** relativa;
- * il **membro del team** che ha il compito di svolgere quel task. La persona deve essere registrata all'interno del *workspace* di i-contact s.r.l.;
- * la **data** entro la quale il compito deve essere svolto. Una settimana prima l'incaricato riceverà una mail di promemoria;
- * uno o più **tags** per differenziare i diversi *tasks* all'interno del progetto.

Una volta assegnato il *task*, incaricato e assegnatario, più tutte le persone che sono in grado di visualizzarlo, potranno aprire una conversazione dedicata nella sezione specifica di quel *task*, dove scambiarsi idee, file e molto altro. Terminato il lavoro, l'incaricato dovrà marcare il *task* come concluso attraverso l'apposita spunta.

²Asana. URL: <https://asana.com/>.

³BeeBole. URL: <https://beebole.com/it/>.

BeeBole BeeBole è uno strumento di *web timesheet* che dà la possibilità ai propri utenti di monitorare in modo efficiente il tempo dedicato a progetti, clienti e incarichi. L'azienda i-contact s.r.l. sfrutta questo applicativo per controllare il budget dedicato ad ogni progetto e le ore effettivamente investite da ogni componente del team, per procedere con una fatturazione corretta e trasparente nei confronti del cliente.

Gestione di versione

L'azienda i-contact s.r.l. utilizza come **sistema di controllo di versione** del codice [Mercurial](#). In particolare, per poter raccogliere tutto il codice derivante dai vari progetti delle varie aree sviluppo, viene utilizzata la versione premium di BitBucket⁴.

Alcuni vantaggi dell'adozione di un sistema di versionamento del codice sono:

- * **ridondanza**: ogni sviluppatore possiede un *backup* della *repository* localmente, limitando così la possibilità di perdita totale dei dati;
- * **disponibilità**: anche in assenza di connessione alla *repository* principale, è possibile continuare ad effettuare *commit* ed a lavorare. Una volta ripristinata la connessione, la *repository* locale può essere sincronizzata con quella remota rendendo le modifiche visibili a tutti;
- * **branch e merge**: permette con molta facilità la creazione dei cosiddetti *branch*, ossia delle ramificazioni dal prodotto stabile presente nella *repository*, per apportare modifiche o nuove funzionalità evitando di introdurre errori e bug. Una volta che anche questa nuova *feature* risulta corretta, può essere inserita all'interno del progetto principale tramite un *merge*.

Comunicazione aziendale

Come strumento di **comunicazione** tra i dipendenti dell'azienda, i-contact s.r.l. ha scelto Slack⁵. Slack è un software che rientra nella categoria degli strumenti di collaborazione aziendale utilizzato per inviare messaggi in modo istantaneo ai membri del team. Il suo punto di forza è la possibilità di creare dei **canali** dedicati a un progetto o ad un particolare argomento, dove vengono inseriti tutti o solo parte dei dipendenti aziendali. È possibile inoltre comunicare con il team anche attraverso chat individuali o chat con due o più membri.

La scelta è ricaduta su questo strumento anche per la sua facilità di utilizzo e la caratteristica di essere fruibile da tutti i dispositivi *iOs*, *Android*, *Windows* come applicazione e da *web browser*.

1.5 Clienti

Grazie ai prodotti offerti da i-contact s.r.l. la sua clientela spazia in molti ambiti diversi come automobilismo, banche, telecomunicazioni e molto altro. Queste aziende si rivolgono a i-contact s.r.l. sia per migliorare le comunicazione tra loro e la propria clientela, attraverso la piattaforma di SMSHosting e le sue numerose soluzioni, sia per proporre lo sviluppo di nuovi prodotti *web based*.

Il team di i-contact s.r.l. ha lavorato, direttamente o tramite i loro partner, con

⁴*BitBucket*. URL: <https://bitbucket.org/>.

⁵*Slack*. URL: <https://slack.com/>.

clienti in tutta Italia. Grazie ad una consolidata modalità di lavoro riescono infatti a garantire ai clienti progetti di qualità, nei tempi previsti e con costi ridotti evitando trasferte e lavorando principalmente dalla loro sede. Alcuni aziende poi si rivolgono a i-contact s.r.l. per consulenze e *partnership* nel ramo dello sviluppo di prodotti dedicati a *smartphone* e *tablet*, dove il team aziendale eccelle particolarmente.

1.6 Struttura del documento

Il documento è stato strutturato per descrivere in maniera esaustiva il percorso di stage nell'azienda i-contact s.r.l.. I capitoli presenti sono i seguenti:

Il primo capitolo descrive l'azienda i-contact s.r.l., a partire dalla sua storia e dai prodotti che essa offre, fino alle tecnologie che vengono utilizzare giornalmente e le metodologie interne adottate nello sviluppo dei prodotti software.

Il secondo capitolo approfondisce il progetto di stage nella sua interezza. Vengono infatti descritti gli strumenti utilizzati dallo stagista, i problemi affrontati, le scelte intraprese e infine viene illustrata una panoramica del prodotto ottenuto durante le 300 ore di lavoro.

Il terzo capitolo approfondisce ...

Il quarto capitolo contiene

Riguardo la stesura del testo, relativamente al documento sono state adottate le seguenti convenzioni tipografiche:

- * gli acronimi, le abbreviazioni e i termini ambigui o di uso non comune menzionati vengono definiti nel glossario, situato alla fine del presente documento;
- * per la prima occorrenza dei termini riportati nel glossario viene utilizzata la seguente nomenclatura: *parola*^[g];
- * i termini in lingua straniera o facenti parti del gergo tecnico sono evidenziati con il carattere *corsivo*.

Capitolo 2

Progetto di stage

L'azienda i-contact s.r.l. ha da sempre mostrato interesse nell'inserimento all'interno del contesto aziendale di stagisti, studenti o altre figure professionali, anche per limitati periodi di tempo. Questa particolare propensione permette all'azienda sia di promuovere nuovi progetti o attività, che con il personale a disposizione non sarebbero percorribili per questioni di tempo, sia introdurre idee e tecnologie innovative per futuri prodotti o per quelli già sviluppati ed utilizzati da i-contact s.r.l..

2.1 Descrizione del progetto

Negli ultimi anni i-contact s.r.l. si è interessata alla creazione e sviluppo di [Chatbot](#) per Facebook Messenger, dando la possibilità ai clienti di richiederne lo sviluppo di nuovi secondo le proprie specifiche, oppure di creare il proprio [Chatbot](#) autonomamente, senza alcuna conoscenza tecnica, attraverso dei template ottimizzati per i più comuni modelli di business.

Il progetto di stage si inserisce in questo ambito, in quanto l'azienda desiderava introdurre la possibilità per gli utenti di scrivere delle vere e proprie domande al [Chatbot](#), senza dover utilizzare il menù o i pulsanti dei vari modelli che la piattaforma mette a disposizione.

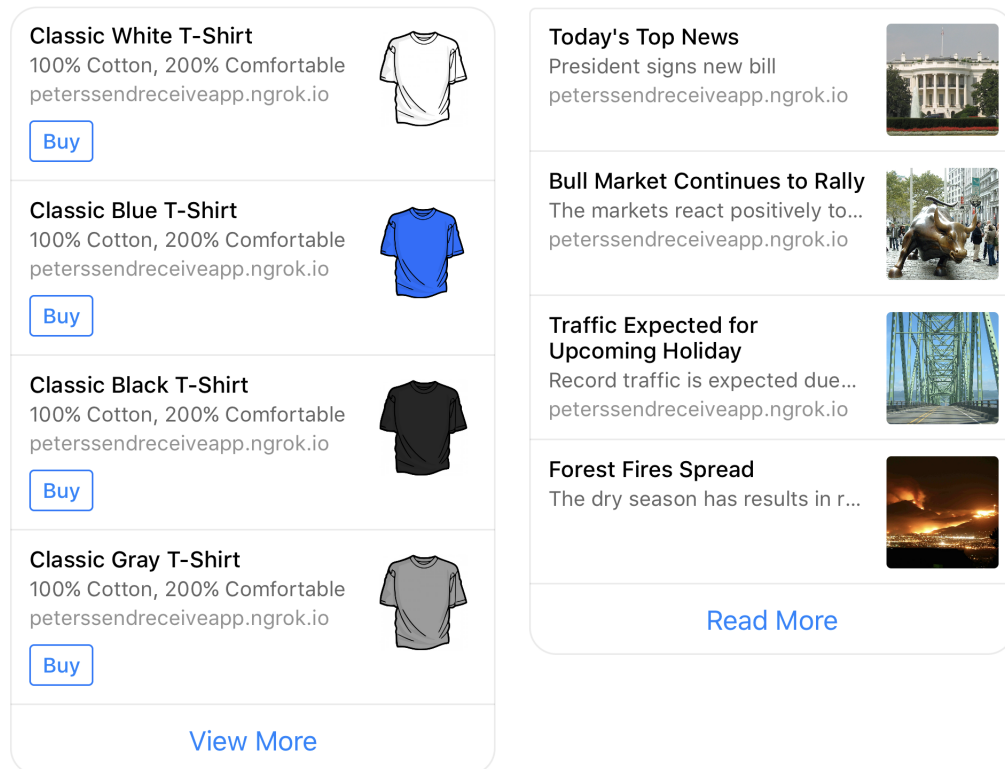


Figura 2.1: Modello generico di carosello per Facebook Messenger

Il compito assegnato allo stagista è stato quello di integrare questo meccanismo in due diversi **Chatbot** creati dall'azienda:

- * **gestore eventi:** è un applicativo creato da i-contact s.r.l. per essere utilizzato durante lo svolgimento di eventi della durata di uno o più giornate. Attraverso questo **Chatbot** l'utente può recuperare delle informazioni riguardanti le conferenze in programma, le aule disponibili e le giornate dell'evento;
- * **meteo Veneto bot**¹ questo **Chatbot** utilizza i dati di ARPA Veneto² per mostrare le previsioni del tempo del Veneto;

Il progetto si divideva quindi in tre parti:

- * analisi preliminare degli SDK delle principali piattaforme per il **NLP** presenti sul mercato, in modo da valutarne pregi e difetti;
- * creazione della logica per la gestione delle domande utente all'interno della piattaforma scelta;
- * integrazione di questa nuova funzionalità nel software utilizzato da i-contact s.r.l. per la gestione dei **Chatbot**.

¹ *Meteo Veneto Bot.* URL: <https://www.smshosting.it/it/blog/chatbot/il-primo-chatbot-sul-meteo-veneto-e-dolomiti>.

² *ARPAV Meteo.* URL: <http://www.arpa.veneto.it/>.

Visto il bisogno di un periodo iniziale di studio, sia per capire le possibilità e i limiti degli strumenti da dover utilizzare nella trasformazione delle [FAQ](#) in dati processabili, sia per apprenderne a pieno il funzionamento tramite la documentazione presente, i-contact s.r.l. ha ritenuto questo progetto idoneo ad uno studente universitario, il quale ha a disposizione circa 300 ore per formarsi su tutto ciò di cui vi è bisogno, e successivamente portare a termine il prodotto richiesto.

2.2 Principali problematiche

Durante il periodo di stage, sono state incontrate diverse difficoltà, utili a migliorare la capacità di scelta e decisione dello stagista, ma anche la comunicazione e il lavoro all'interno di un team.

2.2.1 Difficoltà di ambientazione

Nei primi giorni in azienda l'ostacolo principale è stato l'inserimento in un team di lavoro affiatato e di esperienza, nonostante i suoi componenti siano stati molto disponibili e aperti a domande e chiarimenti. In particolare:

- * il **linguaggio tecnico/aziendale** non è stato da subito di facile apprendimento. In parte il problema sorgerà quando i brainstorming del team vertevano su tecnologie a me sconosciute, oppure su aspetti tecnici dei loro prodotti. Con il passare del tempo, attraverso l'utilizzo di un piccolo glossario individuale dove segnavo le spiegazioni dei termini a me non chiari, ho assimilato la maggior parte dei concetti, potendo così partecipare alle riunioni del team in modo attivo e costruttivo;
- * lavorare all'interno di un team non è un'attività semplice. Bisogna imparare a rispettare le scadenze nel rispetto dei colleghi e saper elaborare strategie nel caso in cui essi non siano riusciti a rispettare le proprie. Grazie all'esperienza che ho maturato sotto questo punto di vista, ho capito quanto sia importante pianificare con cura le attività e il tempo da destinare ad ognuna di esse.

2.2.2 Difficoltà lavorative

Durante lo svolgimento del mio progetto ho riscontrato queste difficoltà principali:

- * scelta del **miglior strumento per il NLP**: il [NLP](#) è un problema che è stato affrontato relativamente recentemente nel mondo dell'informatica. Le tecnologie che il mercato mette a disposizione sono limitate e spesso non soddisfacenti per gli obiettivi che mi sono stati richiesti. La ricerca di uno strumento idoneo non è stata facile, anche perché l'azienda richiedeva non ci fossero costi per il suo utilizzo. Dopo una buona analisi del mercato, con la produzione di una relativa documentazione consegnata al tutor aziendale, è stato deciso di utilizzare [Api.ai](#), servizio acquistato da Google nel 2016, che offre svariate funzionalità senza alcun costo, ma con delle sole limitazioni in termini di richieste giornaliere e mensili (una breve analisi dei principali strumenti di NLP è presente nella sezione??);
- * **possibili richieste degli utenti**: il mio compito era quello di fornire una risposta tramite i [Chatbot](#) alle possibili domande di un utilizzatore. Se in primo momento questo non sembrava un *task* troppo complicato, si è però trasformato

in un problema non di semplice gestione, in quanto il dominio di richieste di un utente può essere vasto e molto vario. Per cercare di superare questo ostacolo, in accordo con il tutor e il team aziendale, si è deciso di limitare le categorie di domande a cui il [Chatbot](#) risponde, fornendo un messaggio adeguato in caso di una richiesta dell'utente non gestita. Sarà poi compito dell'azienda ampliare le funzionalità dei [Chatbot](#), visto il poco tempo a mia disposizione, valutando le richieste fatte dagli utenti nell'apposita sezione di [Api.ai](#);

- * **gestione del sessionId:** nel JSON utilizzato per interrogare [Api.ai](#) tramite una richiesta PUSH, deve essere presente obbligatoriamente il campo `sessionId`, per identificare univocamente il mittente della richiesta. La gestione di questo campo è stata quindi di notevole importanza nel garantire ad ogni utente una corretta interazione con il [Chatbot](#). Per assegnare ad ogni utente un `sessionId` univoco è stato deciso di creare un campo nella tabella degli utenti del database, dove memorizzare la stringa rappresentata dall'id dell'utente nel database e il timestamp del collegamento al bot, garantendo così l'unicità della stessa. In questo modo le domande fatte dall'utente al [Chatbot](#) saranno gestite come una conversazione, invece che una singola domanda senza una storia.

- imparare documentazione - collegamento db java - gestione sessionId - capire tutte possibili domande degli utenti -

2.3 Strumenti utilizzati

2.3.1 NetBeans

NetBeans è un ambiente di sviluppo integrato multi-linguaggio, nato nel giugno 2000 e scritto interamente in Java, scelto dalla Oracle Corporation come IDE ufficiale da contrapporre al più diffuso Eclipse. L'azienda non ha posto nessun vincolo sull'ambiente di sviluppo da adottare, così come vale per i propri dipendenti. Ho deciso di utilizzare NetBeans in quanto è molto intuitivo e di semplice utilizzo.

Inoltre offre una integrazione con [Mercurial](#) per gestire *push*, *pull* e *commit* senza utilizzare le righe di comando.



Figura 2.2: Logo di NetBeans

2.3.2 Api.ai

[Api.ai](#) è una società nata nell'ottobre del 2010 e acquisita da Google Inc. nel 2016. [Api.ai](#) è una piattaforma di conversazione che permette interazioni sofisticate con il linguaggio naturale. All'interno del progetto è stata utilizzata per trasformare le domande degli utenti in dati processabili, dopo aver creato due agenti, uno per ciascun

[Chatbot](#), ed averli istruiti secondo le possibili [FAQ](#) dei rispettivi ambiti di utilizzo. Api.ai poteva essere integrato nel software aziendale in due diversi modi:

- * connesso direttamente al [Chatbot](#) di Facebook Messenger, grazie alla semplice integrazione prevista da Api.ai, estraendo le informazioni dal database aziendale tramite dei *webhook*, anch'essi facilmente gestibili tramite la piattaforma di Api.ai;
- * come strumento esterno al software aziendale, con il bisogno di interrogarlo tramite delle richieste POST HTTP solo in alcuni casi specifici, con le domande poste dall'utente.

Dopo aver valutato entrambe le opzioni, ho scelto di intraprendere la seconda per due motivazioni:

- * utilizzando Api.ai come strumento esterno non è stato necessario introdurre dipendenze verso di esso nel codice, dando così la possibilità, se ci fosse bisogno in futuro, all'azienda di cambiare questa tecnologia con un'altra senza grossi problemi, cosa che sarebbe stata molto più difficile nell'altro caso;
- * Api.ai mette a disposizione un SDK per Java molto utile, anche se non ben documentato. In ogni caso dopo aver capito il suo funzionamento è stato molto semplice sfruttarlo per interrogare Api.ai con poche righe di codice.



Figura 2.3: Logo di api.ai

2.3.3 Hibernate

Hibernate è una piattaforma open-source ad alto rendimento per lo sviluppo di applicazioni Java, che fornisce il servizio Object-relation mapping (ORM), ovvero si occupa della mappatura tra le classi Java e le relative tabelle di un database SQL. Gestisce dunque il salvataggio degli oggetti di tali classi ed il reperimento dalle entità dal database, automatizzando le query necessarie e provvedendo alla reistanziatura dell'oggetto mappato sul database.

2.3.4 Database MySQL

L'azienda i-contact s.r.l. utilizza un database MySQL per la memorizzazione e gestione dei dati relativi ai progetti dei [Chatbot](#). Lo stagista è stato incaricato di capirne la struttura e come esso veniva gestito all'interno dell'applicativo, per poi utilizzarlo al



Figura 2.4: Logo di Hibernate

meglio nello sviluppo del proprio prodotto.

Infine sono stati utilizzati alcuni degli strumenti consolidati nel contesto aziendale, come *BitBucket* e *Mercurial* per quanto riguarda il versionamento del codice, *Slack* per le comunicazioni con gli altri dipendenti e il tutor ed *Asana* per il tracciamento dei *task* da svolgere (strumenti esposti nella sezione [1.4.2](#)).

2.4 Prodotto ottenuto

dgsdg

Glossario

Adaptive Software Development L'*Adaptive Software Development* (ASD) è una metodologia composta da un insieme di regole di sviluppo software inserite in un sistema complessivo detto *Agile Project Management* i cui concetti base sono tre:

- * *Leadership-Collaboration Management* - Uno stile di gestione misto fra gerarchico e collaborativo;
- * *emphFrom Processes to Pattern* - Passaggio dall'idea di processo definito e misurabile a quella di processo non perfettamente definito, quasi un processo *fuzzy*;
- * *Peering into the Future* - Osservazione del futuro per capire come l'idea che produrrà un affare di successo debba essere legata al momento in cui diventerà una forma di business.

. 3, 11

Frequently Asked Questions Le *Frequently Asked Questions*, meglio conosciute con la sigla FAQ, sono letteralmente le "domande poste frequentemente"; più esattamente sono una serie di risposte stilate direttamente dall'autore, in risposta alle domande che gli vengono poste, o che ritiene gli verrebbero poste, più frequentemente dagli utilizzatori di un certo servizio: soprattutto su Internet e in particolare nel web e nelle comunità virtuali vi sono domande ricorrenti alle quali si preferisce rispondere pubblicamente con un documento affinché non vengano poste troppo spesso, in modo da sciogliere i dubbi dei nuovi utenti. 9, 11

File Transfer Protocol in informatica il *File Transfer Protocol* è un protocollo per la trasmissione di dati tra *host* basato su *TCP* e con architettura di tipo client-server. Il protocollo usa connessioni *TCP* distinte per trasferire i dati e per controllare i trasferimenti e richiede autenticazione del client tramite nome utente e password. 2, 11

Software as a service *Software as a service* (SaaS) (Software come servizio in italiano) è un modello di distribuzione del *software* applicativo dove un produttore di *software* sviluppa, opera (direttamente o tramite terze parti) e gestisce un'applicazione web che mette a disposizione dei propri clienti via Internet previo abbonamento. Si tratta spesso, ma non sempre, di un servizio di *cloud computing*. 4, 11

Chatbot in informatica con il termine *Chatbot* si indica un programma che simula una conversazione tra un robot e un essere umano. Questi programmi funzionano o come utenti stessi delle *chat* o come persone che rispondono alle FAQ delle persone che accedono al sito. 2, 7-9, 11

Notifica push La notifica push è una tipologia di messaggistica istantanea con la quale il messaggio perviene al destinatario senza che questo debba effettuare un'operazione di scaricamento (modalità *pull*). Tale modalità è quella tipicamente utilizzata da applicazioni come *WhatsApp*, oppure servizi di sistemi operativi come *Android*, o come numerose applicazioni derivate da siti web (ad esempio il classico servizio meteo o quello delle notizie). Per permettere alle notifiche push di giungere al destinatario è indispensabile che l'applicazione sia attiva, ovvero operi in *background* e sia on-line. Successivamente, occorre che l'utente abbia autorizzato l'applicazione a inviare le notifiche. [2](#), [12](#)

Application Program Interface in informatica con il termine *Application Programming Interface API* (ing. interfaccia di programmazione di un'applicazione) si indica ogni insieme di procedure disponibili al programmatore, di solito raggruppate a formare un set di strumenti specifici per l'espletamento di un determinato compito all'interno di un certo programma. La finalità è ottenere un'astrazione, di solito tra l'hardware e il programmatore o tra software a basso e quello ad alto livello semplificando così il lavoro di programmazione. [2](#), [12](#)

Natural Language Processing in informatica con il termine *Application Programming Interface API* (ing. interfaccia di programmazione di un'applicazione) si indica ogni insieme di procedure disponibili al programmatore, di solito raggruppate a formare un set di strumenti specifici per l'espletamento di un determinato compito all'interno di un certo programma. La finalità è ottenere un'astrazione, di solito tra l'hardware e il programmatore o tra software a basso e quello ad alto livello semplificando così il lavoro di programmazione. [v](#), [8](#), [12](#)