Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA

CORSO DI LAUREA IN INFORMATICA



Editor visuale per la manipolazione di template HTML

Tesi di laurea triennale

Relatore

Prof. Claudio Enrico Palazzi

 ${\it Laure and o}$ Daniele Marin

Anno Accademico 2016-2017

Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage, della durata di trecentoventi ore, dal laureando Daniele Marin presso l'azienda Zucchetti S.p.a.. L'obiettivo di tale attività di stage è l'analisi di varie librerie Javascript per la realizzazione di template HTML, al fine di poter realizzare un editor grafico che permetta la selezione e la modifica dei template per un loro sucessivo inserimento all'interno di pagine HTML. Inoltre è stato effettuato uno studio sul comportamento dei template in ambito responsive, sulla possibilità di inserire plug-in jQuery all'interno dei template e su di un metodo di caricamento delle librerie controllato in modo di non avere più istanze della stessa libreria se utilizzata da diversi template.

Ringraziamenti

Indice

1	Intr	oduzione	1
	1.1	L'azienda	1
		1.1.1 Portal Studio	1
	1.2	Il progetto	2
		1.2.1 Prima parte	2
		1.2.2 Seconda parte	2
2	Lib	erie analizzate	3
	2.1	Considerazioni generali	3
		2.1.1 I template con sintassi mustache	3
	2.2	Mustache.js	3
		2.2.1 Come funziona	3
		2.2.2 Pregi e difetti	3
		2.2.3 Prototipo	3
	2.3	HandlebarsJS	3
		2.3.1 Come funziona	3
		2.3.2 Pregi e difetti	3
		2.3.3 Prototipo	3
	2.4	Ractive.js	3
		2.4.1 Come funziona	3
		2.4.2 Pregi e difetti	3
		2.4.3 Prototipo	3
	2.5	Confronto finale	4
	$\frac{2.6}{2.6}$	Libreria scelta	4
	2.0	Elibicità sectua	4
3		menti e tecnologie legate alla libreria Ractive.js	5
	3.1	Linguaggi utilizzati	5
	3.2	ES2015	5
	3.3	Editors	5
	3.4	Google Chrome Dev Tools	5
	3.5	Flow	5
4	I te	mplate	7
	4.1	Ractive.js	7
		4.1.1 L'oggetto Ractive	7
		4.1.2 Le opzioni principali	7
		4.1.3 La sintassi mustaches	7
		4.1.4 Il two-way data biding	7

viii Indice

		4.1 F CD: 12	_
		4.1.5 Gli eventi	7
		4.1.6 Il virtual DOM	7
		4.1.7 Creazione di un template	7
	4.0	4.1.8 Plug-in di terze parti	7
	4.2	Struttura dei template	7
	4.3	Rendere il template responsive	7
	4.4	Inserimento plug-in jQuery nei template	7
	4.5	Caricamento dei template nelle pagine HTML	7
5	Ana	alisi dei Requisiti	9
	5.1	Applicazione per la modifica dei template	9
		5.1.1 Visualizzazione lista dei template	9
		5.1.2 Visualizzazione template selezionato	9
		5.1.3 Editor per la modifica del template	9
	5.2	Requisiti individuati	9
	5.3	Riepilogo requisiti	9
6	Pro	gettazione	11
Ū	6.1	9	11
	6.2	r	11
	6.3		11
	6.4		11
	6.5		11
7	Res	dizzazione	13
•	7.1		13
	7.2	1	13
	7.3		13
	7.4	1	13
	7.5		13
8	Cor	nclusioni	15
G	8.1		15 15
	0.1	J	15 15
	8.2		15 15
	8.3		15
	0.0	Comoscenze acquisite	10

Elenco delle figure

	1.1	Logo di Zucchetti S.p.a.																											
--	-----	--------------------------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Elenco delle tabelle

Elenco dei frammenti di codice

Introduzione

1.1 L'azienda



Figura 1.1: Logo di Zucchetti S.p.a.

La Zucchetti S.p.a. è una software house con sede a Lodi, che si occupa di soluzioni complete per le aziende, professionisti(commercialisti, consulenti del lavoro, avvocati, curatori fallimentari, notai ecc.) e pubbliche amministrazioni(Comuni, Province, Regioni, Ministeri, società pubbliche ecc.).

Il gruppo Zucchetti è la prima azzienda italiana in Europa con oltre 3300 addetti, 1100 partner e oltre 105000 clienti.

Le soluzioni principali proposte dall'azienda sono :

- Softwere: gestionali, per la sicurezza sul lavoro, analisi business ecc.
- Hardware: per la rilevazione presenze, controllo accessi e controllo produzione.
- Servizi: di outsourcing, cloud computing e data center.

1.1.1 Portal Studio

Lo stage si è svolto nella sede distaccata di Padova che si occupa di ricerca e sviluppo. Tra i software che vengono sviluppati in questa sede è presente Portal Studio che consiste in una WEB application per la creazione di siti web.

L'applicazione offre all'utente un set completo di strumenti per la creazione di pagine web, permette la creazione e modifica in modo grafico della struttura HTML, la gestione degli stili tramite editor grafico per il CSS ed inoltre permette di gestire i dati provenienti da diversi tipi di database, il loro filtraggio e il binding con varie strutture HTML come liste e tabelle.

Il software risulta essere molto maturo e oltre alle funzionalità sopracitate permette

anche la creazione di portlet e pagelet e altri elementi riutilizzabili e la gestione di risorse come dati in formato JSON.

1.2 Il progetto

Il progetto proposto dall'azienda per lo stage, nasce dal desiderio di aggiungere all'applicazione Portal Studio una nuova funzionalità che consiste nell'offrire all'utente la possibilità di inserire nelle proprie pagine HTML dei template già pronti e selezionabili da un insieme prestabilito.

Questo desiderio ha portato l'azienda ad interessarsi ai template engine come Mustache.js, HandlebarJS ecc.

Lo stage si divide in due parti.

La prima consisteva nello studio dei template e degli aspetti ad essi correlati, la seconda nella realizzazione di un editor che ne permettesse la visualizzazione e la modifica.

1.2.1 Prima parte

La prima parte del progetto inizia con la realizzazione di qualche template prototipo, utile sia per studiare le possibilità della libreria scelta sia per avere un insieme di template da inserire nell'editor che è stato realizzato in seguito.

Durante questa parte del progetto l'attenzione è stata rivolta alla possibilità di realizzare template statici, dinamici, template come composizione di altri template (es. lista di contatti) e template che utilizzano SVG.

In seguito alla realizzazione dei template prototipo è stato effettuato uno studio sulla possibilità di rendere i template responsive cioè permetterne la visualizzazione sia su dispositivi desktop che mobile.

La prima parte si è conclusa con uno studio sulla possibilità di realizzare template che contenessero al loro interno plug-in JQuery e sulla gestione del caricamento delle librerie per il loro funzionamento all'interno della pagina HTML in cui vengono inseriti.

1.2.2 Seconda parte

La seconda parte del progetto consisteva nella realizzazione di un editor grafico che permettesse all'utente la selezione di un template da una lista fornita dall'applicazione. In seguito alla selezione del template desiderato quest'ultimo deve essere visualizzato in un box dedicato e viene creato, in base ai dati editabili del template, un editor che ne permette la modifica per vedere come si comporta il template in base alla modifica di vari parametri.

Per determinati template, come quelli considerati composti, l'editor deve dare la possibilità di visualizzare direttamente il JSON contenente i dati e permetterne la modifica.

Non essendo presente una struttura di beck-end proposta dall'applicazione Portal Studio, perché ancora in fase di valutazione, l'editor è stato sviluppato separatamente e il suo model consiste in un insieme di directory, contenenti i vari elementi che compongono i template, suddivise per categoria.

Per il caricamento dei template e dei loro dati vengono utilizzate chiamate http alle varie risorse, non essendo presenti delle API fornite dall'azienda.

Librerie analizzate

- 2.1 Considerazioni generali
- 2.1.1 I template con sintassi mustache
- 2.2 Mustache.js
- 2.2.1 Come funziona
- 2.2.2 Pregi e difetti
- 2.2.3 Prototipo
- 2.3 HandlebarsJS
- 2.3.1 Come funziona
- 2.3.2 Pregi e difetti
- 2.3.3 Prototipo
- 2.4 Ractive.js
- 2.4.1 Come funziona
- 2.4.2 Pregi e difetti
- 2.4.3 Prototipo

- 2.5 Confronto finale
- 2.6 Libreria scelta

Strumenti e tecnologie legate alla libreria Ractive.js

- 3.1 Linguaggi utilizzati
- 3.2 ES2015
- 3.3 Editors
- 3.4 Google Chrome Dev Tools
- 3.5 Flow

I template

- 4.1 Ractive.js
- 4.1.1 L'oggetto Ractive
- 4.1.2 Le opzioni principali
- 4.1.3 La sintassi mustaches
- 4.1.4 Il two-way data biding
- 4.1.5 Gli eventi
- 4.1.6 Il virtual DOM
- 4.1.7 Creazione di un template
- 4.1.8 Plug-in di terze parti
- 4.2 Struttura dei template
- 4.3 Rendere il template responsive
- 4.4 Inserimento plug-in jQuery nei template
- 4.5 Caricamento dei template nelle pagine HTML

Analisi dei Requisiti

- 5.1 Applicazione per la modifica dei template
- 5.1.1 Visualizzazione lista dei template
- 5.1.2 Visualizzazione template selezionato
- 5.1.3 Editor per la modifica del template
- 5.2 Requisiti individuati
- 5.3 Riepilogo requisiti

Progettazione

- 6.1 Suddivisione template
- 6.2 Caricamento template
- 6.3 Creazione lista template
- 6.4 Visualizzazione template selezionato
- 6.5 Editor per la modifica del template

Realizzazione

- 7.1 Il caricamento dei template
- 7.2 Controllo delle librerie caricate
- 7.3 Visualizzatore lista template
- 7.4 Visualizzatore template selezionato
- 7.5 Editor per la manipolazione del template

Conclusioni

In questo capitolo finale vengono tratte le conclusioni riguardo alle attività svolte durante il periodo di stage.

- 8.1 Valutazione del risultato e di Ractive.js
- 8.1.1 Requisiti soddisfatti
- 8.2 Criticità
- 8.3 Conoscenze acquisite