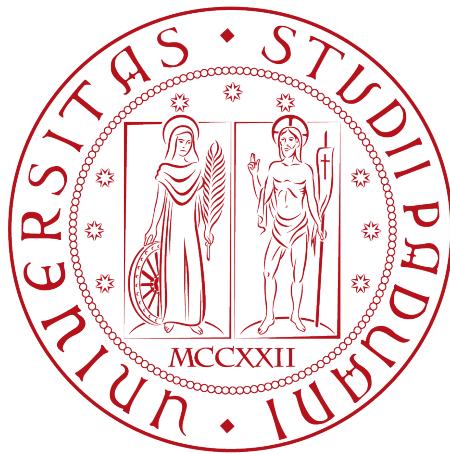


Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA “TULLIO LEVI-CIVITA”

CORSO DI LAUREA IN INFORMATICA



**Ricerca ed analisi comparativa di sistemi di
reportistica centralizzata: alternative a
JasperServer**

Tesi di Laurea Triennale

Relatore

Prof. Marco Zanella

Laureando

Stefano Dal Poz

Matricola 1204683

Contro le difficoltà della vita le armi migliori sono il coraggio, l'ostinazione e la pazienza. Il coraggio fortifica, l'ostinazione diverte e la pazienza dà pace

— Hermann Hesse —

Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage, della durata complessiva di 304 ore, dal laureando Stefano Dal Poz presso l'azienda Bluenext S.R.L. nel periodo compreso tra il 17 novembre 2025 e il 23 Gennaio 2026.

Gli obiettivi da raggiungere erano molteplici.

In primo luogo era richiesta l'analisi di un software di reportistica utilizzato presso l'azienda committente, JasperServer e Jaspersoft Studio, al fine di comprenderne le funzionalità essenziali e per definire i requisiti del nuovo sistema.

In secondo luogo era richiesto lo studio di alcune alternative, sia open source che commerciali, a JasperServer, in particolare BIRT (Business Intelligence and Reporting Tools), con lo scopo di valutarne i pro e i contro rispetto alla soluzione attualmente in uso.

Terzo ed ultimo obiettivo era lo sviluppo di un prototipo di reportistica basato su BIRT, che includesse la creazione di report semplici e complessi, la valutazione della possibilità di integrazione con un sistema di gestione lato server e la generazione di report in diversi formati (PDF, HTML, Excel, DOC).

Ringraziamenti

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

Desidero ringraziare l'azienda Bluenext S.R.L. per avermi dato l'opportunità di svolgere il mio stage presso di loro, in particolare il mio tutor aziendale, Simone Ronchin, per la guida, la pazienza e il supporto dimostratomi durante tutto il periodo dello stage.

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

Ho desiderio di ringraziare poi i miei amici, in particolare Federico e Massimo, per tutte le avventure, i bei momenti vissuti e il sostegno dimostratomi durante questo cammino.

Padova, Aprile 2026

Stefano Dal Poz

Indice

1	Introduzione	1
1.1	L'azienda	1
1.1.1	Prodotti dell'azienda	1
1.2	L'idea	2
1.3	Organizzazione del testo	2
1.3.1	Convenzioni tipografiche	2
2	Descrizione dello stage	3
2.1	Introduzione al progetto	3
2.2	Analisi preventiva dei rischi	3
2.3	Requisiti e obiettivi	4
2.4	Pianificazione	5
3	Analisi dei requisiti	6
3.1	Tracciamento dei requisiti	6
3.1.1	Requisiti funzionali	7
3.1.2	Requisiti non funzionali	7
3.1.3	Requisiti qualitativi	7
3.1.4	Requisiti di vincolo	8
3.1.5	Riepilogo dei requisiti	8
4	Metodologie di analisi e tecnologie analizzate	9
4.1	Studio del prodotto JasperServer	9
4.2	Tecnologie analizzate	9
4.2.1	Birt	9
4.2.2	Dynamic PDF	10
4.2.3	FineReport	10
4.2.4	Helical Insight	11
4.2.5	JsReport	11
4.2.6	Knowage	12
4.2.7	Pentaho	12
4.2.8	ReportServer	13
4.2.9	SAP Crystal Report	13
4.2.10	Stimulsoft	14
5	Progettazione e codifica	15
5.1	Tecnologie e strumenti	15
5.2	Ciclo di vita del software	15
5.3	Progettazione	15
5.4	Design Pattern utilizzati	15
5.5	Codifica	15

6 Prestazioni	16
6.1 Configurazione dei test	16
7 Conclusioni	17
7.1 Consuntivo finale	17
7.2 Raggiungimento degli obiettivi	17
7.3 Conoscenze acquisite	17
7.4 Valutazione personale	17
Acronimi e abbreviazioni	18
Glossario	19
Bibliografia	20

Elenco delle figure

1.1	Logo di Bluenext S.R.L.	1
4.1	Logo di BIRT	10
4.2	Logo di Dynamic PDF	10
4.3	Logo di FineReport	11
4.4	Logo di Helical Insight	11
4.5	Logo di JsReport	12
4.6	Logo di Knowage	12
4.7	Logo di Pentaho	13
4.8	Logo di ReportServer	13
4.9	Logo di SAP Crystal Reports	14
4.10	Logo di Stimulsoft	14

Elenco delle tabelle

2.1	Tabella degli obiettivi	5
3.1	Tabella del tracciamento dei requisiti funzionali	7
3.2	Tabella del tracciamento dei requisiti non funzionali	7
3.3	Tabella del tracciamento dei requisiti qualitativi	8
3.4	Tabella del tracciamento dei requisiti di vincolo	8
3.5	Riepilogo dei requisiti	8

Capitolo 1

Introduzione

In questo capitolo viene presentata l'azienda ospitante, l'idea alla base dello stage e l'organizzazione del testo.

1.1 L'azienda

Il progetto di *stage* di questa tesi è stato svolto presso l'azienda Bluenext Srl, società di consulenza informatica fondata nel 2012 a Rimini e operativa su tutto il territorio nazionale. In particolare, lo *stage* si è svolto presso la sede di Treviso, in una delle filiali distribuite sul territorio italiano, che in passato operava con il nome di SogeaSoft Srl prima di essere acquisita da Bluenext nel 2022.



Figura 1.1: Logo di Bluenext S.R.L.

1.1.1 Prodotti dell'azienda

L'azienda offre soluzioni *software* personalizzate per imprese di varie dimensioni, con l'obiettivo di migliorare i processi aziendali e aumentare l'efficienza operativa. Il target principale è rappresentato dalle Piccole e Medie Imprese (PMI) che desiderano digitalizzare i propri processi e richiedono supporto tecnico specializzato. L'azienda si occupa dello sviluppo di *software* ERP e il suo prodotto di punta è «SAI», un *software* gestionale caratterizzato da una elevata flessibilità che permette di effettuare integrazioni in qualsiasi direzione, rendendolo estremamente adattabile e personalizzabile in base alle specifiche esigenze del cliente.

1.2 L'idea

I sistemi di reportistica centralizzata sono strumenti essenziali per le aziende che necessitano di generare, gestire e distribuire report in modo efficiente. JasperServer è una soluzione ampiamente adottata, ma esistono alternative che potrebbero offrire vantaggi in termini di prestazioni, scalabilità, costi e facilità di integrazione. Questo progetto mira a valutare diverse alternative a JasperServer, analizzandone le caratteristiche, i punti di forza e le potenziali limitazioni arrivando a sviluppare un prototipo basato su una delle soluzioni individuate, al fine di testarne l'efficacia in un contesto reale.

1.3 Organizzazione del testo

Il capitolo corrente è l'introduzione del documento, dove si è spiegato brevemente l'ambito di lavoro e il progetto sul quale si è svolto lo *stage*. In seguito il documento sarà organizzato con la seguente struttura:

Il secondo capitolo approfondisce la descrizione dello stage, illustrando il contesto aziendale, le tecnologie utilizzate e le sfide affrontate durante il periodo di tirocinio.

Il terzo capitolo approfondisce l'analisi dei requisiti del sistema di reportistica, descrivendo il tracciamento dei requisiti.

Il quarto capitolo approfondisce il progetto di *stage*, descrivendo gli obiettivi e le attività svolte, la metodologia di lavoro adottata e le tecnologie analizzate.

Il quinto capitolo è dedicato all'analisi delle soluzioni individuate, alla progettazione e alla codifica del prototipo di reportistica basato su una delle alternative a JasperServer individuate.

Il sesto capitolo descrive le prestazioni ottenute durante i test effettuati sul prototipo di reportistica sviluppato rispetto a JasperServer.

Nel settimo capitolo descrive le conclusioni del lavoro svolto, riassumendo i risultati ottenuti e proponendo possibili sviluppi futuri.

1.3.1 Convenzioni tipografiche

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 parte del gergo tecnico sono evidenziati con il carattere *corsivo*;
- i comandi, le query, i percorsi di file e il codice inline sono rappresentati tramite il carattere *monospaziato*.

Capitolo 2

Descrizione dello stage

In questo capitolo viene presentata la descrizione dello stage, con particolare attenzione al progetto svolto, ai rischi identificati e agli obiettivi raggiunti.

2.1 Introduzione al progetto

Lo *stage* si è svolto presso l'azienda Bluenext Srl, specializzata nello sviluppo di soluzioni software personalizzate per le imprese. L'obiettivo principale dello *stage* è stato quello di ricercare e analizzare sistemi di reportistica centralizzata, con particolare attenzione alle alternative a JasperServer. Il progetto ha previsto una fase di analisi delle soluzioni esistenti, seguita dalla progettazione e implementazione di un prototipo basato su una delle alternative individuate.

2.2 Analisi preventiva dei rischi

L'analisi preventiva dei rischi ha l'obiettivo di identificare possibili criticità tecniche, organizzative e prestazionali, al fine di definire strategie di mitigazione e garantire l'affidabilità della soluzione proposta.

In una prima fase di analisi iniziale sono stati individuati alcuni possibili rischi a cui si potrà andare incontro. Si è quindi proceduto a elaborare delle possibili soluzioni per far fronte a tali rischi.

1. Non conoscenza di *JasperServer*

Descrizione: La mancata conoscenza iniziale di *JasperServer* potrebbe ostacolare la valutazione e la ricerca delle alternative.

Impatto: Alto.

Probabilità: Media.

Soluzione: Per mitigare questo rischio, è stato pianificato uno studio approfondito della documentazione ufficiale di *JasperServer* e l'analisi di casi d'uso reali per comprendere le funzionalità chiave e i requisiti che le alternative devono soddisfare.

2. Integrazione tra componenti eterogenei

Descrizione: L'integrazione tra componenti sviluppati in ambienti e linguaggi differenti (*Java* per il motore di reportistica e *Python* per il server *REST*) potrebbe generare problemi di comunicazione, gestione degli errori o incompatibilità nei formati dei dati scambiati.

Impatto: Alto.

Probabilità: Media.

Soluzione: Definizione di interfacce *REST* standardizzate, utilizzo di formati di scambio dati strutturati (*JSON*) e *test* incrementali delle *API*^[gl] per verificare il corretto funzionamento delle interazioni tra i componenti.

3. Complessità di installazione e configurazione

Descrizione: La complessità di installazione e configurazione delle soluzioni alternative a *JasperServer* potrebbe richiedere tempi e risorse maggiori rispetto alle aspettative iniziali.

Impatto: Medio.

Probabilità: Media.

Soluzione: Redazione di una documentazione tecnica dettagliata delle procedure di installazione e configurazione e utilizzo di container *Docker* per standardizzare l'ambiente di esecuzione.

4. Scalabilità del sistema

Descrizione: La scalabilità del sistema potrebbe essere limitata da fattori tecnologici o architetturali, specialmente in caso di crescita del carico di lavoro o dell'uso simultaneo da parte di molti utenti.

Impatto: Alto.

Probabilità: Media.

Soluzione: Valutazione della scalabilità attraverso *test* di carico e analisi delle architetture supportate. Eventuale separazione tra componente di generazione dei report e componente di esposizione dei risultati.

5. Dipendenza da tecnologie di terze parti

Descrizione: L'utilizzo di *framework* e librerie di terze parti (*JasperReports*, *BIRT Runtime*, *Java*, *Python*) introduce una dipendenza da tecnologie esterne, con possibili problemi di compatibilità, supporto o aggiornamenti futuri.

Impatto: Medio.

Probabilità: Bassa.

Soluzione: Selezione di versioni stabili e ben documentate, verifica della compatibilità tra le dipendenze e definizione di una strategia di aggiornamento controllata.

2.3 Requisiti e obiettivi

Gli obiettivi principali dello *stage* sono stati definiti in base alle esigenze dell'azienda ospitante e alle competenze da sviluppare durante il periodo di tirocinio.

Le fasi specifiche del progetto sono state le seguenti:

- Ricerca e analisi di alternative a *JasperServer* per la reportistica centralizzata.
- Progettazione e implementazione di un prototipo basato su una delle soluzioni individuate.
- Valutazione delle prestazioni del prototipo sviluppato.
- Documentazione dettagliata del progetto e preparazione della presentazione finale.

Ogni fase è stata pianificata con obiettivi specifici e tempistiche definite per garantire il completamento del progetto entro la durata dello *stage*.

Di seguito sono elencati gli obiettivi dello stage:

Tabella 2.1: Tabella degli obiettivi

Obiettivi Obbligatori	
OB1	Studio documentazione e analisi comparativa
OB2	Valutazione e implementazione PoC di due soluzioni alternative
OB3	Documentazione configurazione
Obiettivi Desiderabili	
DE1	Analisi costi e licenze
DE2	Raccolta feedback con tutor o key user da lui individuati
DE3	Definizione best practice di integrazione
Obiettivi Facoltativi	
FA1	Valutazione altre alternative
FA2	Integrazione di funzioni di reporting personalizzato

Per l'analisi dei requisiti si rimanda al Capitolo 3 per definirli in dettaglio e in modo più strutturato.

2.4 Pianificazione

Tutto lo svolgimento dello *stage* è stato pianificato in modo da rispettare le tempistiche stabilite e garantire il raggiungimento degli obiettivi prefissati seguendo la metodologia *Agile*. Le attività previste per lo svolgimento dello *stage* sono state suddivise in cinque fasi principali:

- **Fase 1 - Analisi delle soluzioni esistenti:** ricerca e valutazione delle alternative a JasperServer per la reportistica centralizzata.
- **Fase 2 - Analisi approfondita delle soluzioni individuate:** studio dettagliato delle funzionalità, dei vantaggi e degli svantaggi delle soluzioni selezionate.
- **Fase 3 - Studio specifiche e implementazione del prototipo:** definizione delle specifiche tecniche e sviluppo del prototipo basato sulla soluzione scelta.
- **Fase 4 - Test e valutazione:** esecuzione di *test* per verificare le funzionalità e le performance del prototipo, seguita da una valutazione complessiva.
- **Fase 5 - Documentazione e presentazione:** redazione della documentazione e preparazione della presentazione finale del progetto.

Capitolo 3

Analisi dei requisiti

Questo capitolo presenta l'analisi dettagliata dei requisiti funzionali, qualitativi e di vincolo del progetto.

3.1 Tracciamento dei requisiti

Da un'attenta analisi dei requisiti e degli obiettivi effettuata sul progetto è stata stilata la tabella che traccia i requisiti. Sono stati individuati diversi tipi di requisiti e si è quindi fatto utilizzo di un codice identificativo per distinguerli.

Ogni requisito analizzato sarà identificato univocamente da una sigla del tipo **R[Tipo][Priorità][Codice]** nella quale:

R sta per Requisito;

[Tipo] può essere:

- [F] se Funzionale;
- [NF] se Non Funzionale;
- [Q] se di Qualità;
- [V] se di Vincolo.

[Priorità] può essere:

- [O] per Obbligatorio;
- [D] per Desiderabile;
- [P] per Opzionale.

[Codice] : identifica univocamente i requisiti per ogni tipologia. È un numero intero progressivo univoco assegnato in ordine di importanza se il requisito non ha padre, se invece si tratta di un sotto-requisito segue il formato **[Codice padre].[Numero figlio]** e trattandosi di una struttura ricorsiva non c'è limite alla profondità della gerarchia. I codici sono numerati in base alla sezione e alla classificazione (Es: RFO1 = Requisito Funzionale Obbligatorio 1, RNFO1 = Requisito Non Funzionale Obbligatorio 1, RQO1 = Requisito di Qualità Obbligatorio 1, RVO1 = Requisito di Vincolo Obbligatorio 1).

Nelle tabelle **3.1**, **3.2**, **3.3** e **3.4** sono riassunti i requisiti emersi in fase di analisi, classificati in base alla loro priorità e accompagnati da una breve descrizione della relativa funzionalità.

3.1.1 Requisiti funzionali

I requisiti funzionali descrivono cosa deve fare il sistema. Sono le funzionalità concrete che la soluzione deve offrire per raggiungere gli obiettivi del progetto.

Tabella 3.1: Tabella del tracciamento dei requisiti funzionali

Requisito	Descrizione	Classificazione
RFO1	Il sistema deve fornire un editor grafico per consentire la creazione e la progettazione visuale dei report	Obbligatorio
RFO2	Il sistema deve permettere l'integrazione con servizi e applicazioni che espongono dati tramite <i>API</i> in formato <i>JSON</i>	Obbligatorio
RFO3	Il sistema deve poter gestire report complessi e strutturati in modo modulare, in particolare elementi di sottoreport	Obbligatorio
RFO4	Il sistema deve consentire l'esportazione dei report in molteplici formati (PDF, DOC, XLSX)	Obbligatorio
RFO5	Il sistema deve consentire l'invocazione programmatica dei report da applicazioni esterne tramite <i>API</i>	Obbligatorio
RFO6	Il sistema deve garantire supporto a database <i>SQL</i> e <i>NoSQL</i>	Obbligatorio
RFO7	Il sistema deve fornire la possibilità di personalizzazione e <i>scripting</i> dei report	Obbligatorio

3.1.2 Requisiti non funzionali

I requisiti non funzionali definiscono come il sistema deve comportarsi, cioè le sue proprietà di qualità interna. Non aggiungono nuove funzioni, ma impongono vincoli di prestazioni, sicurezza, disponibilità, scalabilità, affidabilità e manutenibilità.

Tabella 3.2: Tabella del tracciamento dei requisiti non funzionali

Requisito	Descrizione	Classificazione
RNFO1	Il sistema deve supportare la scalabilità, consentendo la gestione di un numero crescente di richieste e utenti	Obbligatorio
RNFO2	Il sistema deve garantire la corretta generazione dei report in modo consistente	Obbligatorio
RNFO3	Il sistema deve consentire una facile integrazione con applicazioni esistenti sviluppate in <i>Java</i> , <i>.NET</i> e <i>Python</i>	Obbligatorio
RNFO4	Il sistema deve garantire tempi di risposta accettabili anche su dataset consistenti	Obbligatorio
RNFO5	Il sistema deve risultare mantenibile, facilitando aggiornamenti, configurazioni e interventi evolutivi	Desiderabile

3.1.3 Requisiti qualitativi

I requisiti qualitativi specificano le proprietà qualitative che influenzano l'esperienza d'uso e la manutenibilità. Si concentrano su aspetti percepibili, come semplicità, chiarezza, flessibilità o estendibilità.

Tabella 3.3: Tabella del tracciamento dei requisiti qualitativi

Requisito	Descrizione	Classificazione
RQO1	Il sistema deve fornire un'interfaccia utente intuitiva e di facile utilizzo	Obbligatorio
RQD2	Il sistema deve garantire coerenza nella progettazione e nel <i>layout</i> dei report	Desiderabile
RQD3	Il sistema deve offrire una buona esperienza di sviluppo, supportata da documentazione chiara e <i>API</i> ben definite	Desiderabile

3.1.4 Requisiti di vincolo

I requisiti di vincolo impongono limitazioni o condizioni esterne al progetto: ambienti, tecnologie, compatibilità, strumenti, standard aziendali o legali.

Tabella 3.4: Tabella del tracciamento dei requisiti di vincolo

Requisito	Descrizione	Classificazione
RVO1	Il sistema deve poter essere installato e gestito su server locali, senza dipendere esclusivamente da soluzioni cloud	Obbligatorio
RVO2	Il sistema deve prevedere condizioni di <i>licensing</i> so-stenibili, evitando vincoli restrittivi legati a licenze proprietarie	Obbligatorio
RVO3	Il sistema deve essere compatibile con l'infrastruttura e lo <i>stack</i> tecnologico aziendale esistente	Desiderabile
RVD4	Il sistema dovrebbe utilizzare tecnologie consolidate e ampiamente supportate dalla <i>community</i> o dal <i>vendor</i>	Desiderabile

3.1.5 Riepilogo dei requisiti

Tabella 3.5: Riepilogo dei requisiti

Tipologia	Obbligatorio	Desiderabile	Opzionale	Totale
Funzionali	7	0	0	7
Non funzionali	4	1	0	5
Qualitativi	1	2	0	3
Vincolo	3	1	0	4
Totale	15	4	0	19

Capitolo 4

Metodologie di analisi e tecnologie analizzate

Questo capitolo descrive le diverse fasi di analisi utilizzate per analizzare un prodotto a partire da JasperServer e le tecnologie utilizzate per lo sviluppo del PoC.

4.1 Studio del prodotto JasperServer

La prima fase del progetto ha previsto uno studio approfondito del prodotto JasperServer, al fine di comprenderne le funzionalità, le caratteristiche tecniche e le modalità di utilizzo per poter . Per analizzare il prodotto JasperServer sono state utilizzate diverse metodologie di analisi, tra cui:

- **Analisi documentale:** è stata effettuata una revisione approfondita della documentazione ufficiale di JasperServer, inclusi manuali utente, guide di installazione e documenti tecnici. Questo ha permesso di comprendere le funzionalità principali del prodotto, le sue caratteristiche tecniche e le modalità di utilizzo.
- **Test pratici:** sono stati eseguiti test pratici utilizzando JasperServer in scenari reali o simulati. Questo ha permesso di valutare le prestazioni del prodotto, la facilità di configurazione e l'efficacia delle funzionalità offerte.

4.2 Tecnologie analizzate

Durante il progetto sono state analizzate diverse tecnologie con lo scopo di individuare quale di queste fosse la più adatta a sostituire JasperServer. Per eseguire questa ricerca sono stati consultati articoli, documentazioni ufficiali e tutorial online.

4.2.1 Birt

Birt (Business Intelligence and Reporting Tools) è una piattaforma open source per la creazione di report e visualizzazioni di dati. È sviluppata dalla Eclipse Foundation ed è ampiamente utilizzata per lo sviluppo di applicazioni di business intelligence. Birt offre una vasta gamma di funzionalità, tra cui la creazione di report interattivi, la generazione di grafici e tabelle, l'integrazione con diverse fonti di dati e la possibilità di personalizzare l'aspetto dei report. La piattaforma Birt è basata su Java e può essere integrata in applicazioni web o desktop. Offre un ambiente di sviluppo intuitivo che consente agli sviluppatori di creare report complessi senza dover scrivere codice complesso. Birt supporta anche l'esportazione dei report in diversi formati, come PDF, Excel, HTML e Word, consentendo agli utenti

di condividere facilmente i report generati. In sintesi, Birt è una piattaforma potente e flessibile per la creazione di report e visualizzazioni di dati, che offre una vasta gamma di funzionalità per soddisfare le esigenze delle applicazioni di business intelligence.



Figura 4.1: Logo di BIRT

4.2.2 Dynamic PDF

Dynamic PDF è una libreria software che consente la generazione dinamica di documenti PDF in modo programmatico. È sviluppata da Ceetron Solutions ed è ampiamente utilizzata per creare report, fatture, etichette e altri documenti PDF personalizzati. La libreria Dynamic PDF offre una vasta gamma di funzionalità, tra cui la creazione di pagine, l'aggiunta di testo, immagini, tabelle e grafici, nonché la formattazione avanzata dei contenuti. Supporta anche l'inserimento di collegamenti ipertestuali, segnalibri e moduli interattivi nei documenti PDF. Dynamic PDF è progettata per essere facile da usare e integrare in applicazioni software esistenti. Offre un'API intuitiva che consente agli sviluppatori di creare documenti PDF in modo rapido ed efficiente, senza dover affrontare la complessità del formato PDF sottostante. In sintesi, Dynamic PDF è una libreria potente e flessibile per la generazione dinamica di documenti PDF, che offre una vasta gamma di funzionalità per soddisfare le esigenze delle applicazioni che richiedono la creazione di documenti PDF personalizzati.



Figura 4.2: Logo di Dynamic PDF

4.2.3 FineReport

FineReport è una piattaforma di business intelligence e reportistica sviluppata da FanRuan Software Co., Ltd. È ampiamente utilizzata per la creazione di report, dashboard e analisi dei dati in ambito aziendale. FineReport offre una vasta gamma di funzionalità, tra cui la creazione di report interattivi, la generazione di grafici e tabelle, l'integrazione con diverse fonti di dati e la possibilità di personalizzare l'aspetto dei report. La piattaforma supporta anche l'esportazione dei report in diversi formati, come PDF, Excel, HTML e Word. FineReport è progettata per essere facile da usare e integrare in applicazioni software esistenti. Offre un ambiente di sviluppo intuitivo che consente agli sviluppatori di creare report complessi senza dover scrivere codice complesso. In sintesi, FineReport è una piattaforma potente e flessibile per la creazione di report e analisi dei dati, che offre

una vasta gamma di funzionalità per soddisfare le esigenze delle applicazioni di business intelligence.



Figura 4.3: Logo di FineReport

4.2.4 Helical Insight

Helical Insight è una piattaforma di business intelligence open source che offre una vasta gamma di strumenti per l'analisi dei dati, la creazione di report e la visualizzazione delle informazioni aziendali. È progettata per essere flessibile e scalabile, consentendo alle organizzazioni di adattare la piattaforma alle proprie esigenze specifiche. Helical Insight offre funzionalità avanzate come l'integrazione con diverse fonti di dati, la creazione di dashboard interattivi, l'analisi predittiva e l'elaborazione in-memory. La piattaforma supporta anche l'esportazione dei report in diversi formati, come PDF, Excel, HTML e Word. Helical Insight è sviluppata in Java e può essere integrata in applicazioni web o desktop. Offre un ambiente di sviluppo intuitivo che consente agli sviluppatori di creare report complessi senza dover scrivere codice complesso. In sintesi, Helical Insight è una piattaforma completa di business intelligence che consente alle organizzazioni di sfruttare al massimo i propri dati per prendere decisioni informate e migliorare le prestazioni aziendali.



Figura 4.4: Logo di Helical Insight

4.2.5 JsReport

JsReport è una piattaforma di reportistica open source che consente la creazione, la gestione e la distribuzione di report in modo semplice ed efficiente. È progettata per essere flessibile e scalabile, consentendo agli sviluppatori di creare report personalizzati utilizzando tecnologie web standard come HTML, CSS e JavaScript. JsReport offre una vasta gamma di funzionalità, tra cui la generazione di report in diversi formati (PDF, Excel, HTML), l'integrazione con diverse fonti di dati, la creazione di dashboard interattivi e la possibilità di programmare l'esecuzione dei report. La piattaforma supporta anche l'uso di modelli personalizzati e l'estensione delle funzionalità tramite plugin. JsReport è sviluppata in Node.js e può essere integrata in applicazioni web o desktop. Offre un'API intuitiva che consente agli sviluppatori di creare report complessi senza dover scrivere codice complesso. In sintesi, JsReport è una

piattaforma potente e flessibile per la creazione di report, che offre una vasta gamma di funzionalità per soddisfare le esigenze delle applicazioni di reportistica.



Figura 4.5: Logo di JsReport

4.2.6 Knowage

Knowage è una piattaforma di business intelligence open source che offre una vasta gamma di strumenti per l'analisi dei dati, la creazione di report e la visualizzazione delle informazioni aziendali. È sviluppata da Engineering Ingegneria Informatica S.p.A. ed è ampiamente utilizzata da organizzazioni di diverse dimensioni per prendere decisioni basate sui dati. Knowage offre una suite completa di strumenti, tra cui Knowage Reporting per la creazione di report, Knowage OLAP per l'analisi multidimensionale dei dati e Knowage Dashboard per la creazione di dashboard interattivi. La piattaforma supporta l'integrazione con diverse fonti di dati, inclusi database relazionali, file di testo, servizi web e big data. Knowage offre anche funzionalità avanzate come l'analisi predittiva, l'elaborazione in-memory e l'integrazione con strumenti di visualizzazione dei dati come Tableau e QlikView. In sintesi, Knowage è una piattaforma completa di business intelligence che consente alle organizzazioni di sfruttare al massimo i propri dati per prendere decisioni informate e migliorare le prestazioni aziendali.



Figura 4.6: Logo di Knowage

4.2.7 Pentaho

Pentaho è una piattaforma di business intelligence open source che offre una vasta gamma di strumenti per l'analisi dei dati, la creazione di report e la visualizzazione delle informazioni aziendali. È sviluppata da Hitachi Vantara ed è ampiamente utilizzata da organizzazioni di diverse dimensioni per prendere decisioni basate sui dati. Pentaho offre una suite completa di strumenti, tra cui Pentaho Data Integration (PDI) per l'estrazione, la trasformazione e il caricamento dei dati (ETL), Pentaho Business Analytics per l'analisi dei dati e la creazione di report, e Pentaho Dashboard Designer per la creazione di dashboard interattivi. La piattaforma supporta l'integrazione con diverse fonti di dati, inclusi database relazionali, file di testo, servizi web e big data. Offre anche funzionalità avanzate come l'analisi predittiva, l'elaborazione in-memory e l'integrazione con strumenti di visualizzazione dei dati come Tableau e QlikView. In sintesi, Pentaho è una piattaforma completa di business intelligence.

che consente alle organizzazioni di sfruttare al massimo i propri dati per prendere decisioni informate e migliorare le prestazioni aziendali.



Figura 4.7: Logo di Pentaho

4.2.8 ReportServer

ReportServer è una piattaforma di reportistica open source che consente la creazione, la gestione e la distribuzione di report in modo semplice ed efficiente. È progettata per essere flessibile e scalabile, consentendo agli sviluppatori di creare report personalizzati utilizzando tecnologie web standard come HTML, CSS e JavaScript. ReportServer offre una vasta gamma di funzionalità, tra cui la generazione di report in diversi formati (PDF, Excel, HTML), l'integrazione con diverse fonti di dati, la creazione di dashboard interattivi e la possibilità di programmare l'esecuzione dei report. La piattaforma supporta anche l'uso di modelli personalizzati e l'estensione delle funzionalità tramite plugin. ReportServer è sviluppata in Java e può essere integrata in applicazioni web o desktop. Offre un'API intuitiva che consente agli sviluppatori di creare report complessi senza dover scrivere codice complesso. In sintesi, ReportServer è una piattaforma potente e flessibile per la creazione di report, che offre una vasta gamma di funzionalità per soddisfare le esigenze delle applicazioni di reportistica.



Figura 4.8: Logo di ReportServer

4.2.9 SAP Crystal Report

SAP Crystal Reports è uno strumento di reportistica sviluppato da SAP che consente agli utenti di creare report interattivi e visivamente accattivanti basati su dati provenienti da diverse fonti. È ampiamente utilizzato in ambito aziendale per generare report dettagliati e analisi dei dati. SAP Crystal Reports offre una vasta gamma di funzionalità, tra cui la creazione di report personalizzati, l'integrazione con diverse fonti di dati (database relazionali, file di testo, servizi web), la generazione di grafici e tavole, e la possibilità di esportare i report

in diversi formati (PDF, Excel, Word). Lo strumento supporta anche funzionalità avanzate come la formattazione condizionale, i sottoreport, le formule personalizzate e l'integrazione con applicazioni .NET e Java. In sintesi, SAP Crystal Reports è uno strumento potente e flessibile per la creazione di report aziendali, che offre una vasta gamma di funzionalità per soddisfare le esigenze delle organizzazioni che necessitano di analisi dei dati e reportistica avanzata.



Figura 4.9: Logo di SAP Crystal Reports

4.2.10 Stimulsoft

Stimulsoft è una suite di strumenti di reportistica e business intelligence che consente agli utenti di creare report interattivi e dashboard personalizzati. È progettata per essere facile da usare e altamente flessibile, supportando diverse fonti di dati e offrendo un'ampia gamma di funzionalità per la generazione e la visualizzazione dei dati. Stimulsoft offre un ambiente visivo intuitivo per la creazione di report, con supporto per diversi formati di output come PDF, Excel, Word e HTML. Inoltre, consente l'integrazione con applicazioni web e desktop, fornendo API e librerie che facilitano l'automazione della generazione dei report. In sintesi, Stimulsoft è uno strumento potente e versatile per la creazione di report aziendali, che offre una vasta gamma di funzionalità per soddisfare le esigenze delle organizzazioni che necessitano di analisi dei dati e reportistica avanzata.



Figura 4.10: Logo di Stimulsoft

Capitolo 5

Progettazione e codifica

Breve introduzione al capitolo

5.1 Tecnologie e strumenti

Di seguito viene data una panoramica delle tecnologie e strumenti utilizzati.

Tecnologia 1

Descrizione Tecnologia 1.

Tecnologia 2

Descrizione Tecnologia 2

5.2 Ciclo di vita del software

5.3 Progettazione

Namespace 1

Descrizione namespace 1.

Classe 1: Descrizione classe 1

Classe 2: Descrizione classe 2

5.4 Design Pattern utilizzati

5.5 Codifica

Capitolo 6

Prestazioni

Questo capitolo descrive le prestazioni ottenute durante i test effettuati sul prototipo di reportistica sviluppato rispetto a JasperServer.

6.1 Configurazione dei test

Capitolo 7

Conclusioni

- 7.1 Consuntivo finale**
- 7.2 Raggiungimento degli obiettivi**
- 7.3 Conoscenze acquisite**
- 7.4 Valutazione personale**

Acronimi e abbreviazioni

API Application Program Interface. 17

Glossario

API 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.

[4](#), [7](#), [8](#), [16](#)

Bibliografia

Riferimenti bibliografici

James P. Womack, Daniel T. Jones. *Lean Thinking, Second Editon.* Simon & Schuster, Inc., 2010.

Siti web consultati

Manifesto Agile. URL: <http://agilemanifesto.org/iso/it/>.