

UNIVERSITÀ DEGLI STUDI DI SALERNO

Dipartimento di Informatica



Laurea Magistrale in Informatica

Software Engineering and IT Management

Disabilità nei Team di Sviluppo Software: una Revisione Sistemica della Letteratura sulle Sfide e sulle Strategie

RELATRICE

Prof.ssa **Filomena Ferrucci**

CORRELATORI

Dott.ssa **Giulia Sellitto**

Dott. **Stefano Lambiase**

CANDIDATA

Giusy Castaldo

Matricola: 0522501106

Anno Accademico 2022-2023

Questa tesi è stata realizzata nel

sesa^{lab}
SOFTWARE ENGINEERING
SALERNO

Alla mia roccia e alla mia luce

AI MIEI GENITORI

Abstract

Lo sviluppo software è uno dei settori in più rapida crescita tuttavia, le persone con disabilità non sono ben rappresentate in tale mondo; da ciò è possibile dedurre che l'industria informatica non fornisca condizioni di partecipazione imparziali e lo sviluppo del software non rifletta le persone che usano la tecnologia. Inoltre, gli studi presenti in letteratura associati alla disabilità nel mondo dello sviluppo software si focalizzano su un unico aspetto, trattandolo in maniera isolata e, dunque, senza considerare l'ambito di interesse nella sua totalità. La revisione e l'aggregazione di tali studi possono contribuire a una migliore comprensione delle persone con disabilità e del loro coinvolgimento nello sviluppo software. Lo scopo del presente lavoro si è incentrato sulla realizzazione di un catalogo di disability smell, corredato dalle refactoring strategy per risolverli. In particolare, l'obiettivo è stato quello di individuare: (1) le ragioni per cui le persone con disabilità lavorano come sviluppatori software, (2) gli attori chiave responsabili di promuovere la partecipazione delle persone con disabilità al mondo dello sviluppo software, (3) le azioni ed iniziative atte a sostenere la partecipazione delle persone con disabilità al mondo dello sviluppo software, (4) le attività predominanti svolte dagli sviluppatori con disabilità, (5) le difficoltà affrontate dagli sviluppatori con disabilità e (6) i modi per affrontare le sfide degli sviluppatori con disabilità con correlati gli attori responsabili di applicarle. La metodologia utilizzata è la systematic literature review per identificare, valutare e interpretare tutta la ricerca disponibile atta a soddisfare le domande di ricerca, tramite l'interrogazione dei più famosi database scientifici.

Indice

Elenco delle Figure	iii
Elenco delle Tabelle	iv
1 Introduzione	1
1.1 Contesto	1
1.2 Limitazioni	3
1.3 Risultati	4
1.4 Struttura dell'elaborato	5
2 Background e stato dell'arte	6
2.1 Background	6
2.1.1 Diversità	6
2.1.2 Disabilità	7
2.1.3 Disabilità in software engineering	9
2.2 Stato dell'arte	10
3 Metodologia di Ricerca	14
3.1 Obiettivo e research question	14
3.2 Formulazione search string	18
3.3 Sorgenti di ricerca	19

3.4	Definizione dei criteri di inclusione/esclusione	20
3.5	Definizione dei criteri di qualità	22
3.6	Estrazione dei dati	23
3.7	Analisi e sintesi dei dati	23
4	Risultati	26
4.1	Esecuzione metodologia di ricerca	26
4.1.1	Applicazione search string	26
4.1.2	Applicazione criteri di inclusione/esclusione	27
4.1.3	Valutazione della qualità	28
4.1.4	Estrazione dei dati	29
4.1.5	Applicazione open coding	30
4.2	Motivazione	30
4.2.1	Ragioni per cui le persone con disabilità lavorano come svilup- patori software	30
4.2.2	Attori chiave responsabili di promuovere la partecipazione delle persone con disabilità	32
4.2.3	Azioni ed iniziative atte a sostenere la partecipazione delle persone con disabilità	33
4.3	Contributo	35
4.3.1	Attività predominanti svolte dagli sviluppatori con disabilità	35
4.4	Sfida	38
4.4.1	Difficoltà affrontate dagli sviluppatori con disabilità	38
4.4.2	Modi per affrontare le sfide degli sviluppatori con disabilità e attori responsabili di applicarle	44
5	Conclusioni	52
5.1	Analisi e sintesi dei risultati	52
	Bibliografia	56
	Ringraziamenti	59

Elenco delle figure

4.1	Risultati ottenuti applicando i criteri di selezione	28
4.2	Focus sui paper esclusi	28

Elenco delle tabelle

1.1	Lavoratori con disabilità, professioni informatiche e matematiche, Stati Uniti	2
1.2	Occupazione 2022, professioni maggiormente svolte, Stati Uniti . . .	2
3.1	Research Question	17
3.2	Search String	20
3.3	Criteri di esclusione	21
3.4	Criteri di inclusione	21
3.5	Criteri di qualità	22
4.1	Risultati quantitativi ottenuti dalla ricerca	27
4.2	Punteggi paper	29
4.3	Paper che rispondono alla domanda di ricerca RQ _{1.1}	31
4.4	Open Coding RQ _{1.1}	31
4.5	Paper che rispondono alla domanda di ricerca RQ _{1.2}	33
4.6	Open Coding RQ _{1.2}	33
4.7	Paper che rispondono alla domanda di ricerca RQ _{1.3}	35
4.8	Open Coding RQ _{1.3}	35
4.9	Paper che rispondono alla domanda di ricerca RQ ₂	37
4.10	Open Coding RQ ₂	38

4.11 Sfide affrontate dagli sviluppatori con disabilità	42
4.12 Paper che rispondono alla domanda di ricerca RQ _{3.1}	43
4.13 Open Coding RQ _{3.1}	44
4.14 Esempio applicazione Voiceye	49
4.15 Paper che rispondono alla domanda di ricerca RQ _{3.2}	51
4.16 Open Coding RQ _{3.2}	51

CAPITOLO 1

Introduzione

Questo capitolo fornisce una panoramica dell'intero lavoro di ricerca, ponendo particolare enfasi sulle motivazioni per cui si è scelto di sviluppare questo lavoro di tesi.

1.1 Contesto

Lo sviluppo software è uno dei settori in più rapida crescita ed ha un impatto diretto sulla vita, sul lavoro e sul tempo libero delle persone. Tuttavia, la mancanza di diversità esistente tra gli sviluppatori involontariamente limita le persone piuttosto che supportarle nel raggiungimento degli obiettivi desiderati. Considerando ciò è possibile dedurre che l'industria informatica non fornisca condizioni di partecipazione imparziali e lo sviluppo del software non rifletta le persone che usano la tecnologia [1].

Ciò si evince in particolar modo tenendo conto di alcune statistiche: negli Stati Uniti, secondo il National Bureau of Labor Statistics¹, nell'anno 2022, rispetto alla percentuale totale di lavoratori che vive con una disabilità, solo il 3% circa è impe-

¹<https://www.bls.gov/news.release/disabl.t03.htm>

gnato in professioni informatiche e matematiche. Considerando lo stesso ambito professionale la percentuale di lavoratori che vive senza una disabilità, è del 4%.

Tabella 1.1: Lavoratori con disabilità, professioni informatiche e matematiche, Stati Uniti

Totale	Uomini	Donne
3.2%	4.0%	2.3%

Per comprendere la percentuale del 3% nella tabella seguente è possibile effettuare un confronto con le mansioni maggiormente svolte dai lavoratori con disabilità.

Tabella 1.2: Occupazione 2022, professioni maggiormente svolte, Stati Uniti

Occupazione	Totale	Uomini	Donne
Occupazioni dirigenziali	11.5%	13.2%	9.7%
Occupazioni nelle vendite	9.2%	7.8%	10.7%
Occupazioni d'ufficio e di supporto amministrativo	11.3 %	5.9 %	17.3
Occupazioni nel trasporto di materiali	9.8 %	14.0 %	5.3%

In aggiunta, ponendo come esempio la disabilità visiva, secondo le stime del 2019 del Piano d'azione globale dell'Organizzazione Mondiale della Sanità, in tutto il mondo sono presenti 285 milioni di persone affette da disabilità visive (4% della popolazione mondiale), di cui: [2]:

- 19 milioni sono bambini con età inferiore ai quindici anni;
- 12 milioni sono adolescenti e giovani adulti con età compresa tra i quindici ed i ventiquattro anni;
- 254 milioni sono adulti dai venticinque anni e oltre.

Nello stesso anno, circa 1.350 dei 90.000 intervistati (1,5%) al sondaggio annuale per sviluppatori di Stack Overflow² si è identificato come affetto da disabilità visiva [3, 4].

Da ciò è possibile dedurre quanto sia basso il tasso di persone con disabilità che lavorano nell'ambito informatico e quante siano le difficoltà incontrate.

²<https://stackoverflow.com/>

La tecnologia non è neutra bensì è influenzata da chi la realizza, pertanto considerare il punto di vista delle persone con disabilità può migliorare la creazione di software inclusivi. Inoltre, il mondo dello sviluppo software soffre spesso di una carenza di personale adeguatamente formato. Includere individui con disabilità può comportare un impatto positivo sia da punto di vista sociale che economico. Da ciò è possibile concludere che vi è una reale necessità di strumenti, processi e formazione che consentano a tutti di accedere nel campo dello sviluppo software affinché quest'ultimo sia diversificato e inclusivo.

1.2 Limitazioni

La maggior parte degli sforzi sulla diversità nel campo dello sviluppo software si è incentrata sulle donne e sulle minoranze, ponendo poca attenzione alla disabilità in tale ambito.

Inoltre, gli studi presenti in letteratura associati alla disabilità nel mondo dello sviluppo software si focalizzano su un unico aspetto, trattandolo in maniera isolata e, dunque, senza considerare l'ambito di interesse nella sua totalità.

Di seguito sono riportati alcuni esempi, per cui ciascuno studio si focalizza su uno dei seguenti aspetti:

- Accessibilità degli strumenti di prototipazione ad alta fedeltà;
- Accessibilità e miglioramento degli ambienti di programmazione;
- Esperienze educative dei programmatori con disabilità;
- Utilizzo dell'interazione vocale multimodale per programmatori con menomazioni fisiche;
- Sfide affrontate dai programmatori con disabilità visive;
- Comprensione dell'accessibilità e della collaborazione nella programmazione per persone con disabilità visive.

La ricerca, dunque, risulta essere spesso disconnessa, con studi isolati e non collegati l'uno all'altro.

1.3 Risultati

A seguito delle limitazioni evidenziate, la revisione e l'aggregazione degli studi presenti in letteratura, associati alla ricerca in esame, possono contribuire a una migliore comprensione delle persone con disabilità e del loro coinvolgimento nello sviluppo software.

Attraverso la mappatura sistematica della letteratura è possibile identificare gli studi pertinenti e raccogliere informazioni da essi.

I risultati di questi studi possono essere utilizzati per definire meccanismi adeguati che favoriscano la partecipazione delle persone con disabilità nel contesto dello sviluppo software, possono portare alla produzione di nuova conoscenza e a una visione più completa della seguente area di studio.

In sintesi, questo studio si propone di raccogliere, rivedere e sintetizzare la ricerca esistente sulla partecipazione delle persone con disabilità nel mondo dello sviluppo software, al fine di comprendere meglio la loro esperienza e definire approcci più inclusivi e adeguati. L'esecuzione dell'analisi della letteratura trova riscontro in un foglio di lavoro online disponibile e accessibile [5].

Il contributo del seguente studio include la caratterizzazione delle persone con disabilità nel contesto dello sviluppo software in particolare, quali sono:

- Le ragioni per cui le persone con disabilità lavorano come sviluppatori software;
- Gli attori chiave responsabili di promuovere la partecipazione delle persone con disabilità al mondo dello sviluppo software;
- Le azioni ed iniziative atte a sostenere la partecipazione delle persone con disabilità al mondo dello sviluppo software;
- Le attività predominanti svolte dagli sviluppatori con disabilità;
- Le difficoltà affrontate dagli sviluppatori con disabilità;
- I modi per affrontare le sfide degli sviluppatori con disabilità con correlati gli attori responsabili di applicarle.

1.4 Struttura dell'elaborato

Il seguente lavoro di tesi è strutturato come segue:

- Capitolo 2.** Verranno introdotti i concetti fondamentali alla base del lavoro in esame e sarà effettuata una descrizione di Systematic Literature Review relative all'ambito di ricerca corrente;
- Capitolo 3.** Il fulcro del lavoro di tesi sarà rappresentato attraverso una descrizione formale, includendo il design dello studio e i vari passaggi che verranno seguiti;
- Capitolo 4.** L'attenzione sarà focalizzata sull'esecuzione pratica delle fasi definite in modo teorico nel terzo capitolo. I dati estratti dagli studi analizzati e le analisi condotte verranno utilizzati per rispondere alle domande di ricerca;
- Capitolo 5.** Si andranno a riassumere le principali conclusioni della ricerca.

CAPITOLO 2

Background e stato dell'arte

In questo capitolo verranno introdotti i concetti fondamentali alla base del lavoro in esame e sarà effettuata una descrizione di Systematic Literature Review relative all'ambito di ricerca corrente.

2.1 Background

2.1.1 Diversità

Diversità si riferisce alla varietà di rappresentazioni che esistono all'interno di un gruppo, basate su una vasta gamma di sfaccettature e caratteristiche le quali la rendono multidimensionale [1].

Tra queste individuiamo la diversità dei valori (ad esempio convinzioni, obiettivi), la diversità delle informazioni (ad esempio esperienza, conoscenza, background) e la diversità sociale (ad esempio genere, età) [6].

Le problematiche associate alla diversità riscontrate nei gruppi di lavoro possono essere spiegate dalla:

- Teoria della somiglianza-attrazione, formulata da Byrne in "The Attraction Paradigm" [7], la quale postula che gli individui che lavorano in gruppo preferiscono

interagire con persone simili a loro [6].

- Prospettiva di categorizzazione sociale, sviluppata da Turner in “Rediscovering the social group: A self-categorization theory” [8], la quale definisce che quando diverse dimensioni della diversità (come il genere, le differenze di personalità o gli atteggiamenti e i valori) convergono, si crea una linea di frattura della diversità che può suscitare la categorizzazione di sottogruppi, una distinzione “noi-loro”. Quest’ultima può interrompere i processi di gruppo rendendo i membri meno fiduciosi e motivati a cooperare tra di loro, aumentando le tensioni e i conflitti interpersonali, e l’abbassamento della comunicazione. Pertanto, sebbene la diversità possa stimolare le prestazioni del gruppo, può anche minarle attraverso processi di categorizzazione sociale [9].

Molte aziende sono consapevoli della mancanza e dell’importanza di diversità nelle loro organizzazioni e questo ha portato ad un’ondata di sforzi per aumentarla. In effetti, negli ultimi decenni la diversità è stata sempre più riconosciuta come una caratteristica essenziale di una squadra. Ad esempio, senza diversità di nazionalità, i team potrebbero non beneficiare di molteplici punti di vista, disponibilità di conoscenze, abilità e conflitti costruttivi [6].

Tuttavia, mentre notevoli sforzi sono stati compiuti per affrontare le difficoltà sia delle donne che delle minoranze, una classe ancora poco esplorata è relata alla disabilità.

2.1.2 Disabilità

La convenzione delle Nazioni Unite sui diritti delle persone con disabilità, Articolo 1 Comma 2, definisce le persone con disabilità come “coloro che presentano durature menomazioni fisiche, mentali, intellettive o sensoriali che, con la presenza di barriere di diversa natura, possono ostacolare la loro piena ed effettiva partecipazione nella società” [10].

Quindi, si possono definire le seguenti categorie [11]:

- **Disabilità fisica:** una persona fisicamente disabile presenta una condizione che limita una o più attività fisiche come camminare, salire le scale, sollevare

o trasportare. Questo tipo di disabilità comprende la disabilità motoria: difficoltà a muovere gli arti nello svolgimento delle attività della vita quotidiana. La disabilità fisica oltre a poter essere visibile (immediatamente evidente) o invisibile (meno evidente), può essere progressiva (in cui lo stato di salute e il funzionamento dell'individuo peggiorano nel tempo) o stabile (in cui lo stato di salute e il funzionamento non cambiano). La disabilità fisica progressiva può essere causata da condizioni come la sclerosi multipla, la distrofia muscolare e l'artrite cronica, mentre alcune condizioni associate a disabilità fisica stabile includono paralisi cerebrale, spina bifida e lesioni del midollo spinale.

- **Disabilità mentale:** questa è una condizione che influenza il normale funzionamento della mente evidenziata da cambiamenti comportamentali. Può essere causata da: schizofrenia, disturbi dell'umore (ad esempio depressione, ipomania), disturbi d'ansia (ad esempio disturbo ossessivo-compulsivo, fobie specifiche, ecc.), disturbi dell'alimentazione (ad esempio bulimia, anoressia), disturbi della personalità (ad esempio disturbo narcisistico di personalità, ecc.), disturbi cerebrali organici (ad esempio ictus, ecc.) e così via;
- **Disabilità intellettiva:** si riferisce alla capacità intellettuale significativamente inferiore alla media della popolazione. È determinata quantitativamente utilizzando il quoziente intellettivo. Si può dire che un individuo con un quoziente inferiore a 70 abbia una disabilità intellettiva. Può essere causata da qualsiasi condizione che colpisce lo sviluppo cerebrale prima della nascita, durante la nascita o negli anni dell'infanzia (età 3-11). Può anche essere dovuta da una malattia, dall'uso di droghe o alcol della madre durante il periodo di gravidanza;
- **Disabilità visiva:** questa è una compromissione della vista di un individuo. Circa il 10% delle persone con disabilità visive è totalmente cieco, mentre circa il 90% è parzialmente cieco;
- **Disabilità uditiva:** si tratta di una menomazione dell'udito. Le disabilità uditive possono essere causate da complicazioni durante la gravidanza o il parto, infezioni, malattie genetiche, farmaci, rumori forti e così via;

- **Disturbo dell'apprendimento:** è definito come un disturbo specifico e persistente del sistema nervoso centrale di un individuo che influisce sul suo processo di apprendimento. Dunque, si riferisce a una serie di condizioni che influenzano specifiche abilità di apprendimento, come la lettura, la scrittura e il calcolo matematico e ostacolano il normale processo di acquisizione delle abilità scolastiche. Un indicatore comune della disabilità dell'apprendimento è la differenza tra il potenziale di un individuo (attitudine e capacità intellettuale) e i suoi effettivi risultati. I disturbi specifici dell'apprendimento non hanno origine in una patologia ma in una diversa modalità di sviluppo neuronale che limita alcune abilità cognitive.

Il 15% della popolazione mondiale vive con qualche forma di disabilità [11].

2.1.3 Disabilità in software engineering

Come evidenziato, sebbene necessario, la maggior parte degli sforzi sulla diversità si è concentrata sulle donne e sulle minoranze, ma poco è stato realizzato in merito alla disabilità. In particolare, si evince che l'Informatica sia uno dei settori in più rapida crescita [12] e, al tempo stesso, non sia un campo notevolmente diversificato. Per questa ragione, gli sforzi per la diversità nell'Informatica iniziano a muovere i primi passi per promuovere l'accessibilità per gli sviluppatori con disabilità e riconoscerli come un gruppo sottorappresentato [13].

In tal senso, svolge un ruolo significativo World Wide Web Consortium¹ (W3C), un'organizzazione internazionale che sviluppa standard per il web. L'organizzazione incoraggia la formazione di gruppi di lavoro inclusivi e offre opportunità per contribuire allo sviluppo di standard e specifiche che riguardano l'accessibilità. Inoltre, supporta la ricerca e l'innovazione nel campo dell'accessibilità web. Collabora con università e centri di ricerca per sviluppare nuove soluzioni e tecnologie che migliorino l'accessibilità degli strumenti e delle piattaforme di sviluppo.

Tra le iniziative proposte dalla W3C rientra Web Accessibility Initiative (WAI), progetto che si occupa di promuovere l'accessibilità delle tecnologie web per persone con disabilità. L'obiettivo principale del WAI è fornire un insieme di standard e criteri

¹<https://www.w3.org/>

tecniche utili sia per gli sviluppatori con disabilità che per gli sviluppatori che vogliono garantire l'accessibilità dei loro prodotti.

La principale linea guida sviluppata dal WAI è chiamata Web Content Accessibility Guidelines (WCAG) e, ad oggi, la versione più recente è la 2.1, pubblicata nel 2018.

Le WCAG sono strutturate in quattro principi di accessibilità:

1. **Perceivable**: si riferisce alla necessità di rendere i contenuti web accessibili a tutti, indipendentemente dalle capacità sensoriali;
2. **Operable**: riguarda l'usabilità e la navigabilità dei siti web per tutti gli utenti;
3. **Understandable**: si riferisce alla chiarezza e alla comprensibilità dei contenuti web;
4. **Robust**: si riferisce alla compatibilità dei contenuti web con una vasta gamma di tecnologie e dispositivi.

Sulla base delle linee guida dell'accessibilità, come WCAG, gli strumenti di valutazione dell'accessibilità web (Web Accessibility Evaluation Tools) analizzano il contenuto e la struttura di un sito web al fine di identificare eventuali problemi o barriere che potrebbero impedire alle persone con disabilità di accedere al sito. Gli strumenti di valutazione dell'accessibilità web possono essere suddivisi in due categorie principali:

- **Strumenti automatici**: eseguono una scansione automatica del codice e dei contenuti del sito web per individuare possibili problemi di accessibilità, basandosi su algoritmi e criteri predefiniti;
- **Strumenti manuali**: richiedono un'analisi umana approfondita da parte di esperti, che valutano l'accessibilità del sito web tenendo conto di aspetti che gli strumenti automatici potrebbero non rilevare in modo accurato.

2.2 Stato dell'arte

In questa sezione viene effettuata un'analisi delle Systematic Literature Review presenti nell'ambito trattato dalla ricerca corrente.

Per comprendere al meglio il lavoro che si andrà a svolgere si esamina un ambito correlato ma distinto dalla seguente ricerca, in quanto associato allo sviluppo di software accessibile. Lo studio proposto da Paiva et al., “Accessibility and Software Engineering Processes: A Systematic Literature Review” [14], presenta i risultati significativi che delineano come le proposte di metodi, tecniche, processi, approcci e risorse per supportare lo sviluppo di applicazioni accessibili sono progredite nell'ultimo decennio.

Si riporta, in effetti, un leggero aumento del numero di pubblicazioni dal 2011 al 2019.

Da tale studio è emerso che i principali linguaggi di programmazione utilizzati per implementare strumenti e framework sono Java, HTML, JavaScript, Android, XML e PHP. In relazione, invece, ai modelli utilizzati si individua WCAG, analizzata nel dettaglio precedentemente, che si è affermata come riferimento primario per quanto riguarda le linee guida sull'accessibilità e lo sviluppo di strumenti di valutazione dell'accessibilità.

In termini di diversità nell'Ingegneria del Software, gli studi ad essa associati sono importanti per comprendere la disparità che si verifica oggi nei luoghi di lavoro nell'ambito della tecnologia dell'informazione. Lo studio di Menezes et al., “Diversity in Software Engineering” [15], ha come obiettivo l'analisi della diversità nell'Ingegneria del Software e definisce come comportarsi quando abbiamo team con caratteristiche differenti. Le domande di ricerca affrontate in tale studio sono:

- Quali studi sono stati pubblicati sulla diversità nell'Ingegneria del Software e quali sono le loro caratteristiche?
- In che modo la diversità influisce sul processo di sviluppo del software?

In merito alla prima domanda di ricerca, sono stati individuati diversi studi che affrontano argomenti associati alla diversità. I punti salienti della maggior parte di essi riguardano le barriere e le sfide da superare soprattutto nel lavoro di squadra. Inoltre, emerge che le disparità hanno inizio nell'ambiente accademico con corsi legati all'Informatica che non attraggono gruppi di minoranza. Ciò si riflette anche nel mercato del lavoro.

Relativamente alla seconda domanda di ricerca, è stato osservato che le pratiche adottate negli ambienti di lavoro possono contribuire all'inclusione, oltre che a migliori risultati organizzativi. La diversità potrebbe avere un impatto positivo su aspetti come l'efficienza dell'innovazione e il lavoro di squadra nelle diverse fasi di sviluppo del software.

Rispetto alla ricerca su cui si base il seguente lavoro, tale studio risulta avere un carattere più generalizzante e, non focalizzato esclusivamente sulle disabilità nell'Ingegneria del Software.

Nel contesto della diversità rientra, per l'appunto, la disabilità, tra cui individuiamo quella visiva. Lo studio proposto da Mountapmbeme et al., "Addressing Accessibility Barriers in Programming for People with Visual Impairments: A Literature Review" [16], ha come obiettivo l'identificazione e l'aggregazione delle barriere di accessibilità affrontate da programmatori professionisti e studenti con disabilità visive che imparano a programmare, nonché identificare tutte le soluzioni che sono state proposte per affrontare queste barriere. Dunque, sono state individuate le seguenti domande di ricerca:

- Quali sono le barriere di accessibilità note alla programmazione per le persone con disabilità visive?
- Quali sono le soluzioni esistenti e proposte a queste barriere?

Relativamente alla prima domanda di ricerca, le sfide affrontate dalle persone con disabilità visive nella programmazione includono sfide legate alla navigazione, alla comprensione, alla modifica, al debug e alla scrematura del codice. Si evidenzia che queste sfide siano una conseguenza della natura degli strumenti utilizzati dalle persone con disabilità visive.

In merito alla seconda domanda di ricerca, le soluzioni esistenti associate alle sfide dei programmatori con disabilità visiva includono plug-in e strumenti di accessibilità, nuovi linguaggi di programmazione accessibili e kit di strumenti di programmazione accessibili che fanno uso di molteplici modalità di interazione progettate per studenti alle prime armi con disabilità visive.

È necessario sottolineare che, al giorno d'oggi, l'attenzione si concentra sulla disabilità visiva completa o parziale e, pochi studi discutono altre disabilità, come le disabilità uditive e cognitive [14].

Sulla base di quanto analizzato è possibile dedurre che ciascuno studio pone l'attenzione su un particolare aspetto della disabilità nell'Ingegneria del Software. Ciò che manca è una Systematic Literature Review che possa fornire un quadro completo del campo, includendo aspetti associati ai problemi che gli sviluppatori con disabilità affrontano sul lavoro così da categorizzarli per poter ottenere proposte di soluzioni adatte a supportare tali sviluppatori.

CAPITOLO 3

Metodologia di Ricerca

In questo capitolo si andrà a rappresentare il fulcro del lavoro di tesi in cui si dettaglierà in modo formale ciò che è stato pianificato andando a specificare il design della ricerca ed i vari step del seguente studio.

3.1 Obiettivo e research question

Al fine di ampliare gli sforzi associati alla diversità per supportare i professionisti dell'Informatica con disabilità, è importante comprendere le sfide che devono superare [13].

Questo lavoro si concentra sulla realizzazione di un catalogo di disability smell, corredato dalle refactoring strategy per risolverli.

Per comprendere entrambi i concetti effettuiamo un'analogia con quanto accade nel contesto del software.

Analogia. Gli sviluppatori modificano, migliorano e adattano continuamente il software e, man mano che evolve, quest'ultimo si allontana dal suo design originale.

Ciò comporta:

1. Diminuzione della comprensibilità dovuta ad una maggiore complessità del codice, ad una documentazione obsoleta e a codice non conforme agli standard;
2. Minore affidabilità;
3. Aumento dei costi di manutenzione.

Per ridurre la complessità del software è necessario migliorarne la qualità interna effettuando una ristrutturazione, cioè riorganizzando il software (codice sorgente e documentazione) per dargli un aspetto o una struttura diversa. Quindi, la ristrutturazione del software è informalmente indicata come le modifiche del software per renderlo più facile da capire, più facile da cambiare (anche in termini di documentazione), meno suscettibile ai guasti quando vengono apportate modifiche.

La ristrutturazione applicata ad un software orientato agli oggetti è chiamata refactoring [17].

Per ristrutturare un sistema software si segue un processo con attività ben definite [18]:

1. Rilevare parti di codice con opportunità di refactoring;
2. Determinare quale refactoring può essere applicato nel frammento di codice in analisi;
3. Assicurarci che il refactoring selezionato preservi il comportamento;
4. Applicare il refactoring scelto;
5. Valutare l'effetto del refactoring sulle caratteristiche di qualità del software;
6. Mantenere la coerenza tra l'artefatto sottoposto a refactoring e altri artefatti software.

Il programmatore identifica gli elementi su cui effettuare il refactoring da un set di artefatti software di alto livello [17]:

- Codice sorgente;
- Documenti di progettazione;
- Documenti dei requisiti.

Successivamente, ci si concentra su parti specifiche dell'artefatto scelto per il refactoring. È possibile identificare moduli, funzioni, classi, metodi e dati specifici. In generale, le tecniche di refactoring sono altamente specializzate, con una tecnica che migliora un piccolo numero di attributi di qualità. Per esempio, alcuni refactoring eliminano la duplicazione del codice, alcuni aumentano la riutilizzabilità, alcuni migliorano le prestazioni, alcuni migliorano la manutenibilità [17].

Il termine "smell" si riferisce ad un problema interno al software, sia ad un livello inferiore, noto come livello di codice, sia ad un livello più alto, di design, che descrive i sintomi che compromettono l'evoluzione del software osservati nelle componenti. A differenza di un bug, uno smell non provoca necessariamente un fault nell'applicazione, ma può portare ad altre conseguenze negative, impattando sulla manutenzione e sull'evoluzione del software. Sulla base del livello a cui ci si riferisce, viene rispettivamente definito come code smell o design smell [18].

Concentrandoci sul concetto di code smell, questo viene applicato al codice sorgente per rilevare ciò che deve essere sottoposto a refactoring. Un code smell è qualsiasi sintomo nel codice sorgente che potrebbe indicare un problema più profondo.

Esempi sono [17]:

- Codice duplicato;
- Lungo elenco di parametri;
- Metodi lunghi;
- Classi molto grandi.

In conclusione, il refactoring rappresenta l'approccio principale per la rimozione di smell, aiuta a rendere il codice più leggibile e ad eliminare possibili problemi, oltre a migliorare la qualità interna ed esterna del software [18]. Alcuni esempi di qualità interne del software sono dimensione, complessità, accoppiamento, coesione e testabilità. Alcuni esempi di qualità esterne del software sono prestazioni, riutilizzabilità, manutenibilità, estensibilità, robustezza e scalabilità [17].

Da ciò se ne evince che, così come esistono i code smell e il refactoring del codice, i problemi di inclusione possono essere definiti come disability smell e risolti applicando strategie di refactoring.

Quindi, il seguente lavoro si sofferma su quali siano i problemi che gli sviluppatori con disabilità affrontano sul lavoro così da poter individuare le strategie per promuovere e sostenere la partecipazione degli sviluppatori con disabilità ai progetti software.

Inoltre, si andrà a comprendere cosa motiva gli sviluppatori con disabilità a contribuire ai progetti e quali tipologie di contributo forniscono.

La ricerca è stata guidata dall'analisi di cinque research question, categorizzate in base a motivazione, contributo e sfida:

Tabella 3.1: Research Question

#	Domande	Categorie
1.1	Cosa motiva le persone con disabilità a lavorare come sviluppatori software?	Motivazione
1.2	Chi dovrebbe incentivare e sostenere la partecipazione delle persone con disabilità al mondo dello sviluppo software?	Motivazione
1.3	Cosa si può fare per sostenere la partecipazione delle persone con disabilità al mondo dello sviluppo software?	Motivazione
2	Quali sono le attività maggiormente svolte da sviluppatori con disabilità?	Contributo
3.1	Quali difficoltà gli sviluppatori con disabilità si trovano ad affrontare?	Sfida
3.2	Quali sono le strategie e i metodi per affrontare tali sfide? Chi deve applicarle?	Sfida

La metodologia utilizzata è la systematic literature review per identificare, valutare e interpretare tutta la ricerca disponibile atta a soddisfare le domande di ricerca in esame. I singoli studi che contribuiscono a una revisione sistematica sono chiamati studi primari; una revisione sistematica è una forma di studio secondario. La revisione sistematica della letteratura inizia, quindi, definendo le research question. Successivamente si attua una strategia di ricerca definita che mira a individuare quanta più letteratura pertinente possibile per poi utilizzare criteri di inclusione ed esclusione per valutare ogni potenziale studio primario. Fondamentale è specificare le informazioni da ottenere da ciascuno studio primario [19].

3.2 Formulazione search string

La fase cardine per l'individuazione di articoli correlati all'argomento di tesi consiste nel ricercare keywords che consentano di ottenere una search string da cui far partire la ricerca. Nel contesto in esame è stata considerata inizialmente, come unica parola chiave, "developer with disabilities". Per poter ottenere un numero cospicuo di studi presenti in letteratura associati alla ricerca in esame, sono stati individuati sinonimi quali: "developer with disabilities", "programmer with disabilities", "developer with impairments", "programmer with impairments", "impaired programmer", "impaired developer", "disabled programmer", "disabled developer". Sono stati, inoltre, aggiunti i plurali di quanto precedentemente individuato: "developers with disabilities", "programmers with disabilities", "developers with impairments", "programmers with impairments", "impaired programmers", "impaired developers", "disabled programmers", "disabled developers".

La search string risultante, dunque, è la seguente: "developer with disabilities" OR "programmer with disabilities" OR "developer with impairments" OR "programmer with impairments" OR "impaired programmer" OR "impaired developer" OR "disabled programmer" OR "disabled developer" OR "developers with disabilities" OR "programmers with disabilities" OR "developers with impairments" OR "programmers with impairments" OR "impaired programmers" OR "impaired developers" OR "disabled programmers" OR "disabled developers".

È possibile notare come, essendo la search string composta da soli sinonimi, questa sia realizzata esclusivamente utilizzando l'operatore booleano OR.

3.3 Sorgenti di ricerca

A seguito dalla fase di definizione della search string, quest'ultima è stata applicata a tre database:

- Scopus¹
- IEEE Xplore²
- ACM Digital Library³

Ciò ha consentito di condurre una revisione sistematica della letteratura per poter ottenere un set di articoli pertinenti alla ricerca in esame. Per effettuare la ricerca, la search string è stata adattata al database di riferimento nel seguente modo:

¹<https://www.scopus.com/>

²<https://ieeexplore.ieee.org/>

³<https://dl.acm.org/>

Tabella 3.2: Search String

Database	Stringhe di ricerca
Scopus	TITLE-ABS-KEY ((developer with disabilities) OR {programmer with disabilities} OR {developer with impairments} OR {programmer with impairments} OR {impaired programmer} OR {impaired developer} OR {disabled programmer} OR {disabled developer} OR {developers with disabilities} OR {programmers with disabilities} OR {developers with impairments} OR {programmers with impairments} OR {impaired programmers} OR {impaired developers} OR {disabled programmers} OR {disabled developers})
IEEE Xplore	("All Metadata": "developer with disabilities") OR ("All Metadata": "programmer with disabilities") OR ("All Metadata": "developer with impairments") OR ("All Metadata": "programmer with impairments") OR ("All Metadata": "impaired programmer") OR ("All Metadata": "impaired developer") OR ("All Metadata": "disabled programmer") OR ("All Metadata": "disabled developer") OR ("All Metadata": "developers with disabilities") OR ("All Metadata": "programmers with disabilities") OR ("All Metadata": "developers with impairments") OR ("All Metadata": "programmers with impairments") OR ("All Metadata": "impaired programmers") OR ("All Metadata": "impaired developers") OR ("All Metadata": "disabled programmers") OR ("All Metadata": "disabled developers")
ACM Digital Library	[All: "developer with disabilities"] OR [All: "programmer with disabilities"] OR [All: "developer with impairments"] OR [All: "programmer with impairments"] OR [All: "impaired programmer"] OR [All: "impaired developer"] OR [All: "disabled programmer"] OR [All: "disabled developer"] OR [All: "developers with disabilities"] OR [All: "programmers with disabilities"] OR [All: "developers with impairments"] OR [All: "programmers with impairments"] OR [All: "impaired programmers"] OR [All: "impaired developers"] OR [All: "disabled programmers"] OR [All: "disabled developers"]

3.4 Definizione dei criteri di inclusione/esclusione

Una volta ottenuti gli studi primari potenzialmente rilevanti, è necessario valutarne la reale rilevanza. I criteri di selezione hanno lo scopo di identificare quegli studi

primari che forniscono prove dirette alle domande di ricerca [19]. Tali criteri vengono categorizzati in:

- Criteri di esclusione, tali per cui se ne deriva l'esclusione dello studio in analisi dal set di articoli se è soddisfatto almeno uno di essi;
- Criteri di inclusione, tali per cui se ne deriva l'inclusione dello studio in analisi dal set di articoli se è soddisfatto almeno uno di essi.

Di seguito sono riportate le tabelle contenenti i criteri considerati:

Tabella 3.3: Criteri di esclusione

#	Criteri di esclusione
CE1	L'articolo non è correlato all'argomento di interesse
CE2	L'articolo non è interamente disponibile gratuitamente/non è reperibile
CE3	L'articolo non è recente (più di 7 anni)
CE4	Workshop, poster
CE5	L'articolo contiene meno di 7 pagine
CE6	L'articolo non è scritto in inglese
CE7	L'articolo è un duplicato

Tabella 3.4: Criteri di inclusione

#	Criteri di inclusione
CI1	L'articolo riporta cosa motiva le persone con disabilità a lavorare come sviluppatori software
CI2	L'articolo riporta come e chi dovrebbe incentivare la partecipazione delle persone con disabilità al mondo dello sviluppo software
CI3	L'articolo riporta quali sono le attività maggiormente svolte da sviluppatori con disabilità
CI4	L'articolo riporta le difficoltà che si trovano ad affrontare gli sviluppatori con disabilità
CI5	L'articolo riporta quali sono e chi deve applicare le strategie e i metodi per affrontare le sfide degli sviluppatori con disabilità

3.5 Definizione dei criteri di qualità

La fase successiva al filtraggio è associata alla qualità di ogni paper in esame, definita in base alla pertinenza degli obiettivi di ricerca rispetto all'argomento trattato, alla metodologia di ricerca utilizzata e alla disponibilità di informazioni dettagliate per rispondere alle domande di ricerca.

Sono stati definiti sette criteri di qualità, categorizzati in base a:

- Ricerca;
- Raccolta dei dati;
- Contesto;
- Risultati e conclusioni.

Tabella 3.5: Criteri di qualità

#	Criteri di qualità	Categorie
CQ1	L'obiettivo della ricerca è ben spiegato e motivato?	Ricerca
CQ2	La metodologia di ricerca è sufficientemente spiegata?	Ricerca
CQ3	Le Research Question sono chiaramente definite?	Ricerca
CQ4	La metodologia per la raccolta dei dati è descritta in modo sufficientemente dettagliato?	Raccolta dei dati
CQ5	Le informazioni preziose per la revisione sono chiaramente riportate e indicate?	Contesto
CQ6	Vengono discusse le minacce alla validità?	Risultati e conclusioni
CQ7	I risultati sono riportati chiaramente?	Risultati e conclusioni

A ciascun articolo è stato associato un valore numerico (0, 0.5, 1) tale per cui:

- 0 indica che l'articolo non ha rispettato il criterio di qualità;
- 0.5 indica che l'articolo ha discretamente rispettato il criterio di qualità;
- 1 indica che l'articolo ha completamente rispettato il criterio di qualità.

Il conferimento di un punteggio a ciascun paper, definito in base alla somma dei singoli valori numerici associati a ciascun criterio di qualità, non è stato attribuito con la finalità di applicare un processo di selezione dei paper nella revisione sistematica bensì, esclusivamente di stabilire una gerarchia degli studi in base alla qualità metodologica, alla rilevanza e accuratezza dei risultati rispetto agli obiettivi della ricerca.

3.6 Estrazione dei dati

Una volta identificati i paper finali da considerare per tale lavoro, si è partiti dalle domande di ricerca definite andando successivamente ad estrarre, da ciascun paper in esame, dati che fossero utili a rispondere ad una specifica domanda di ricerca.

Dunque, sono state analizzate le informazioni rilevanti per la ricerca, quali: il nome del paper, i nomi degli autori, cosa motiva le persone con disabilità a lavorare come sviluppatori software, come e chi dovrebbe incentivare la partecipazione delle persone con disabilità al mondo dello sviluppo software, quali sono le attività maggiormente svolte da sviluppatori con disabilità, quali sono le difficoltà che si trovano ad affrontare gli sviluppatori con disabilità, quali sono e chi deve applicare le strategie e i metodi per affrontare le sfide degli sviluppatori con disabilità.

3.7 Analisi e sintesi dei dati

In tale sezione si va a chiarire come le informazioni raccolte nella fase di estrazione dei dati siano state sintetizzate, combinate e analizzate per rispondere alle domande di ricerca.

Per fare ciò la strategia utilizzata è l'Open Coding [20]:

- La parola Open si riferisce a mantenere una mente aperta quando si analizzano i dati, cioè non essere influenzati dalla letteratura precedente e/o dalle esperienze personali;
- La parola Coding si riferisce al compito di analisi dei dati.

Tale strategia sta nel concettualizzare e articolare gli aspetti di una serie di estratti che sono risultati rilevanti durante la lettura attenta dei singoli studi. Dunque, è fatta per costruire un insieme di concetti e approfondimenti basati sugli estratti dei documenti [21].

Per spiegare al meglio tale procedura applicata al seguente studio, definiamo alcuni concetti chiave [20]:

- I dati grezzi rappresentano le informazioni raccolte durante la ricerca dai documenti analizzati;
- I punti chiave sono i punti riassuntivi del dato grezzo;
- I codici rappresentano il primo livello di astrazione e sono frasi che riassumono il punto chiave in due o tre parole;
- I concetti sono il secondo livello di astrazione dato dal raggruppamento, usando il metodo del confronto costante, di più codici appartenenti allo stesso documento o ad altri documenti analizzati;
- La categoria è il livello più alto di astrazione dato dal raggruppamento di tutti i concetti.

Dunque, le fasi che hanno caratterizzato l'analisi e la sintesi dei dati, nel seguente studio, possono essere così definite:

1. Per ogni domanda di ricerca si individuino i paper che rispondono a tale domanda (facilmente individuabili dalla fase precedente di Data Extraction);
2. Prendendo in esame una specifica domanda di ricerca, per ogni paper ad essa associato:
 - (a) Si consideri la risposta fornita;
 - (b) Considerando tale risposta, si definiscano i punti chiave;
 - (c) Da ogni punto chiave, si estrarrebbero i codici;
3. Conclusasi la fase 2 (fase iterativa in quanto applicata per ogni paper associato alla domanda di ricerca in analisi), usando il metodo del confronto costante

tra i codici, si raggruppino questi ultimi per produrre un livello più alto di astrazione, il concetto;

4. Si individui, in conclusione, la categoria che consenta di accorpare tutti i concetti.

CAPITOLO 4

Risultati

In questo capitolo si andrà a porre particolare attenzione sull'esecuzione delle fasi definite teoricamente nel precedente capitolo, per poi utilizzare i dati estratti dagli studi analizzati e le analisi effettuate per rispondere alle domande di ricerca del seguente lavoro, definite sulla base dell'obiettivo da raggiungere nel campo dello sviluppo software.

4.1 Esecuzione metodologia di ricerca

Successiva alla definizione delle fasi fondamentali della revisione sistematica della letteratura, è stata affrontata l'esecuzione della stessa.

4.1.1 Applicazione search string

L'esecuzione della search string sui tre database selezionati (Scopus¹, IEEE Xplore² e ACM Digital Library³) ha comportato l'acquisizione di un totale di 81 risultati.

¹<https://www.scopus.com/>

²<https://ieeexplore.ieee.org/>

³<https://dl.acm.org/>

Tabella 4.1: Risultati quantitativi ottenuti dalla ricerca

Scopus	IEEE Xplore	ACM Digital Library
23	4	54

Successiva all'individuazione di tali risultati, candidati ad essere selezionabili per rispondere alle domande di ricerca del lavoro in esame, ciascuno è stato riportato in un foglio di lavoro online [5] con le proprie informazioni rilevanti:

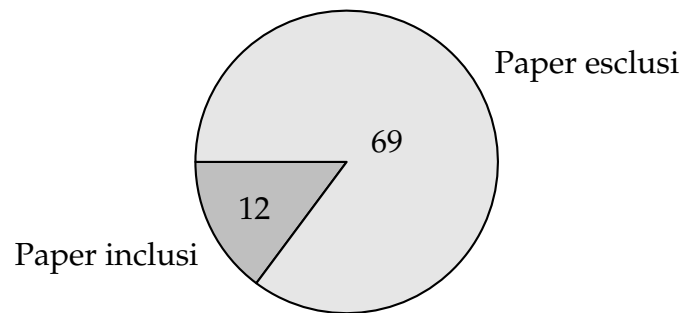
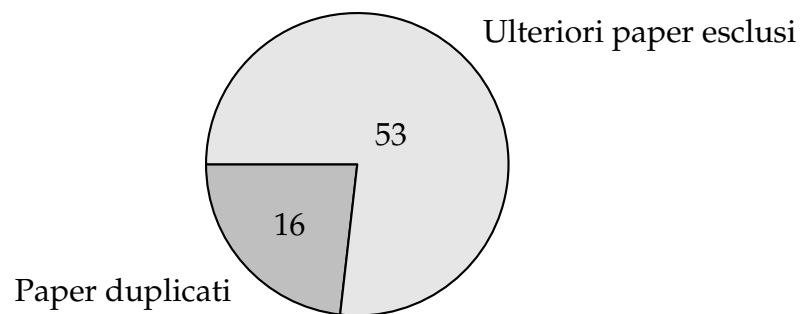
- Titolo;
- Anno di pubblicazione;
- Abstract;
- Link per reperirlo;
- Autori;
- Parole chiave;
- Pagine;
- Tipo di documento.

4.1.2 Applicazione criteri di inclusione/esclusione

Sugli 81 risultati, ottenuti dell'esecuzione della search string, sono stati applicati i criteri di inclusione ed esclusione.

Da ciò ne è risultato che:

- 69 paper hanno rispettato almeno uno dei criteri di esclusione ed, in particolare, 16 paper sono risultati duplicati;
- 12 paper hanno rispettato almeno uno dei criteri di inclusione.

Figura 4.1: Risultati ottenuti applicando i criteri di selezione**Figura 4.2:** Focus sui paper esclusi

Quanto sopra definito ha comportato l'aggiornamento del foglio di lavoro [5], tale per cui:

- I paper duplicati sono stati evidenziati col colore giallo;
- I paper esclusi sono stati evidenziati col colore rosso;
- I paper inclusi sono stati evidenziati col colore verde.

4.1.3 Valutazione della qualità

Successiva alla fase preliminare di filtraggio, è stata effettuata la valutazione della qualità relativamente ad ogni paper risultante incluso dall'applicazione dei criteri di inclusione/esclusione con i punteggi ottenuti per ciascun criterio di qualità, come mostrato nella tabella 4.2.

Tabella 4.2: Punteggi paper

Paper	CQ1	CQ2	CQ3	CQ4	CQ5	CQ6	CQ7
"PSST: Enabling Blind or Visually Impaired Developers to Author Sonifications of Streaming Sensor Data" [22]	0.5	0.5	0	0.5	0.5	0	0.5
"Tangible music programming blocks for visually impaired children" [4]	1	0.5	1	0.5	1	0	1
"Accessible AST-based programming for visually-impaired programmers" [23]	1	1	0	0.5	1	0	0.5
"Educational experiences of blind programmers" [13]	1	1	0	0.5	1	0	0.5
"CodeTalk: Improving programming environment accessibility for visually impaired developers" [12]	1	0.5	0	0.5	1	0	1
"Node-Read: A Visually Accessible Low-Code Software Development Extension" [24]	1	1	1	0.5	1	1	0.5
"Voiceye: A Multimodal Inclusive Development Environment" [25]	1	1	0	1	1	0	1
"StoryBlocks: A Tangible Programming Game To Create Accessible Audio Stories" [26]	1	0.5	1	0.5	0.5	0	0.5
"Inclusive Multimodal Voice Interaction for Code Navigation" [27]	1	0.5	0	0.5	1	0	1
"Usability, Accessibility and Social Entanglements in Advanced Tool Use by Vision Impaired Graduate Students" [28]	1	1	0	1	1	0	1
"Accessibility of High-Fidelity Prototyping Tools" [29]	1	0.5	0	0.5	1	0	1
"Understanding Accessibility and Collaboration in Programming for People with Visual Impairments" [3]	1	1	1	0.5	1	0	1

4.1.4 Estrazione dei dati

La fase di estrazione dei dati è stata fondamentale per rispondere alle domande di ricerca. Le informazioni associate ad ogni paper, per maggiore comprensione e reperibilità, sono state riportate nel foglio di lavoro [5] precedentemente citato.

Nelle tabelle 4.3, 4.5, 4.7, 4.9, 4.12 e 4.15 è possibile, inoltre, prendere visione dei paper che forniscono informazioni rilevanti associate ad una specifica domanda di ricerca e che, dunque, rispondono a tale domanda.

4.1.5 Applicazione open coding

I dati acquisiti dalla fase precedente sono stati meticolosamente esaminati e sintetizzati al fine di fornire risposte accurate per ciascuna domanda di ricerca. Quanto affermato è riportato nelle sezioni successive, con ciascuna domanda di ricerca associata ad un preciso obiettivo.

In particolar modo le tabelle 4.4, 4.6, 4.8, 4.10, 4.13 e 4.16 mostrano informazioni quali codici, concetti e categoria associati a ciascuna domanda di ricerca.

4.2 Motivazione

4.2.1 Ragioni per cui le persone con disabilità lavorano come sviluppatori software

RQ_{1.1}. Cosa motiva le persone con disabilità a lavorare come sviluppatori software?

Negli ultimi anni c'è stato un **crescente incoraggiamento** fin dall'infanzia per le persone con disabilità ad acquisire competenze di programmazione, a tal punto da far sì che quest'ultima sia diventata parte integrante nei curriculum delle scuole primarie [4].

Questo approccio promuove un ambiente di apprendimento inclusivo e dimostra alle persone con disabilità che le loro abilità e capacità sono apprezzate, generando un senso di fiducia e determinazione.

Anche ricevere un **supporto appropriato** è fondamentale in quanto l'entusiasmo provocato da ciò si trasforma in una motivazione interna a perseguire la propria passione per la programmazione e a superare gli ostacoli che si possono incontrare lungo il percorso [4].

Il supporto può essere inteso come l'**accesso a strumenti tecnologici adeguati** [13]: la tecnologia inclusiva può eliminare le barriere che potrebbero altrimenti limitare la partecipazione delle persone con disabilità nel settore dello sviluppo software. Questi strumenti offrono loro la possibilità di lavorare in modo autonomo, affrontare sfide complesse e contribuire con idee innovative. Sapendo che hanno a disposizione

gli strumenti adeguati per svolgere il loro lavoro in modo efficiente, le persone con disabilità sono motivate a perseguire una carriera come sviluppatori software.

Ciò si evince in particolar modo in “Tangible music programming blocks for visually impaired children” [4] di Sabuncuoglu et al. in cui, al termine dello studio sugli utenti improntato sull’utilizzo di una piattaforma musicale tangibile che mira a insegnare le basi della programmazione attraverso la creazione di musica, la maggior parte dei partecipanti ha chiesto come continuare a programmare, quali sarebbero stati i prossimi aggiornamenti e come avrebbero potuto apprendere nuovi linguaggi di programmazione. Tutti i partecipanti hanno affermato di essersi sentiti incoraggiati ad iniziare ad apprendere la programmazione grazie alla loro piacevole esperienza con tale piattaforma. Anche coloro che conoscevano la codifica basata su testo, hanno dichiarato che l’utilizzo di un linguaggio appositamente progettato per loro è stato molto entusiasmante [4].

Nel perseguire la loro passione per la programmazione, le persone con disabilità stanno rompendo gli stereotipi e dimostrando che le abilità e le competenze non conoscono limiti.

Tabella 4.3: Paper che rispondono alla domanda di ricerca RQ_{1.1}

Domanda di ricerca	Paper
RQ _{1.1}	“Tangible music programming blocks for visually impaired children” [4], “Educational experiences of blind programmers” [13]

Tabella 4.4: Open Coding RQ_{1.1}

Codici	Concetti	Categoria
Incoraggiare l’apprendimento della programmazione fin da bambini	Avere incoraggiamento	Motivazione
Suscitare entusiasmo nell’utilizzo di supporti adeguati	Avere entusiasmo	Motivazione
Incoraggiare tramite la presenza di strumenti adeguati	Avere incoraggiamento	Motivazione

4.2.2 Attori chiave responsabili di promuovere la partecipazione delle persone con disabilità

RQ_{1.2}. Chi dovrebbe incentivare e sostenere la partecipazione delle persone con disabilità al mondo dello sviluppo software?

La partecipazione delle persone con disabilità al mondo dello sviluppo software dovrebbe essere incentivata e sostenuta da diverse parti interessate.

Di seguito sono elencati gli attori chiave che svolgono un ruolo fondamentale:

- **Personale educativo** [4, 13, 26]: nell'ambito scolastico, svolge un ruolo fondamentale nel promuovere la partecipazione delle persone con disabilità. L'inclusione scolastica è un concetto chiave che mira a garantire che tutti gli studenti, indipendentemente dalle loro capacità, abbiano accesso a un'educazione di qualità, in questo caso nel contesto informatico. Inoltre, è fondamentale che gli insegnanti siano adeguatamente formati per comprendere le esigenze specifiche degli studenti con disabilità;
- **Comunità di sviluppatori** [28]: può svolgere un ruolo attivo nell'incentivare e sostenere le persone con disabilità. In particolar modo, in un contesto lavorativo, i collaboratori svolgono un ruolo essenziale nel facilitare la partecipazione delle persone con disabilità. In effetti, fungono da supporto funzionale: ciò si riferisce all'assistenza pratica necessaria per garantire che le persone con disabilità possano svolgere le loro mansioni in modo efficace. Ad esempio, una persona con disabilità visiva potrebbe aver necessità di descrizioni verbali di immagini prive di testo alternativo;
- **Aziende del settore tecnologico** [28]: giocano un ruolo fondamentale nell'incentivare la partecipazione delle persone con disabilità;
- **Organizzazioni di supporto** [13]: occupandosi di disabilità dovrebbero lavorare per sensibilizzare e sostenere la partecipazione delle persone con disabilità nel settore dello sviluppo software.

La collaborazione tra tutte queste parti interessate può contribuire a rimuovere le barriere, a favorire l'autonomia e a creare opportunità concrete per le persone con

disabilità, permettendo loro di partecipare attivamente nel campo dello sviluppo software.

Tabella 4.5: Paper che rispondono alla domanda di ricerca RQ_{1.2}

Domanda di ricerca	Paper
RQ _{1.2}	“Tangible music programming blocks for visually impaired children” [4], “Educational experiences of blind programmers” [13], “StoryBlocks: A Tangible Programming Game To Create Accessible Audio Stories” [26], “Usability, Accessibility and Social Entanglements in Advanced Tool Use by Vision Impaired Graduate Students” [28]

Tabella 4.6: Open Coding RQ_{1.2}

Codici	Concetti	Categoria
Sostenere la partecipazione a partire dall'ambito scolastico	Scuola	Contesto
Sostenere la partecipazione a partire dai contesti educativi	Scuola	Contesto
Essere sostenuti (studenti) da insegnanti e personale educativo	Scuola	Contesto
Supporto da parte dei collaboratori	Comunità	Contesto
Supporto aziendale	Aziende	Contesto
Supporto da parte di organizzazioni	Organizzazioni di supporto	Contesto

4.2.3 Azioni ed iniziative atte a sostenere la partecipazione delle persone con disabilità

RQ_{1.3}. Cosa si può fare per sostenere la partecipazione delle persone con disabilità al mondo dello sviluppo software?

Per sostenere la partecipazione delle persone con disabilità al mondo dello sviluppo software è importante considerare diversi aspetti e adottare diverse strategie.

In primo luogo, è necessario condurre **ricerche approfondite** per identificare i bisogni specifici delle persone con disabilità nel contesto dello sviluppo software [4]. Ciò implica la valutazione degli ostacoli che possono incontrare e l'analisi delle sfide che affrontano [4, 12].

Tali informazioni possono essere utilizzate per guidare la progettazione inclusiva, identificare gli strumenti e le tecnologie accessibili, e sviluppare soluzioni mirate per affrontare le sfide incontrate [4, 12].

Fondamentale, quindi, è effettuare un' **evoluzione costante delle tecnologie** utilizzate nello sviluppo software per migliorare il supporto fornito alle persone con disabilità [25].

È altresì importante creare **opportunità di apprendimento e formazione**, in termini di insegnamento della codifica e delle competenze necessarie per lavorare nel settore informatico, per le persone con disabilità che desiderano partecipare al mondo dello sviluppo software [4].

Ciò può essere attuato tramite l'offerta di corsi, workshop o programmi di formazione accessibili e inclusivi, che si adattino alle diverse esigenze degli individui con disabilità e che includano, dunque, l'uso di strumenti e tecnologie assistive, la fornitura di materiali didattici accessibili e l'adattamento delle modalità di insegnamento alle diverse abilità [13, 4]. Ad esempio, AccessComputing⁴ ha tentato di aumentare la partecipazione delle persone con disabilità nell'informatica coltivando attività di tutorato e sostenendo finanziariamente la partecipazione a conferenze e stage [13].

Occorre promuovere l' **interazione sociale e la collaborazione** quindi, favorire l'inclusione di persone con disabilità nei team di sviluppo e incoraggiare la partecipazione attiva di tutti i membri [13, 3]. Importante è che i programmatori con disabilità e i loro colleghi co-creino nuove pratiche di lavoro per collaborare in modo efficace. Le pratiche sono anche modellate alle caratteristiche del team, alla difesa e al lavoro dei programmatori con disabilità [3].

⁴<https://www.washington.edu/accesscomputing/>

Tabella 4.7: Paper che rispondono alla domanda di ricerca RQ_{1.3}

Domanda di ricerca	Paper
RQ _{1.3}	“Tangible music programming blocks for visually impaired children” [4], “Educational experiences of blind programmers” [13], “CodeTalk: Improving programming environment accessibility for visually impaired developers” [12], “Voiceye: A Multimodal Inclusive Development Environment” [25], “Understanding Accessibility and Collaboration in Programming for People with Visual Impairments” [3]

Tabella 4.8: Open Coding RQ_{1.3}

Codici	Concetti	Categoria
Formulare ricerche	Comprendere le necessità/Sfide	Sostegno
Derivare considerazioni di progettazione	Garantire accessibilità	Sostegno
Insegnare la codifica	Fornire insegnamenti	Sostegno
Conoscere gli ostacoli che incontrano	Comprendere le necessità/Sfide	Sostegno
Strumenti accessibili	Garantire accessibilità	Sostegno
Favorire l’interazione sociale	Avere collaborazione	Sostegno
Analizzare le sfide affrontate	Comprendere le necessità/Sfide	Sostegno
Individuare delle soluzioni	Garantire accessibilità	Sostegno
Consentire l’evoluzione delle tecnologie	Garantire accessibilità	Sostegno
Collaborare per realizzare nuove pratiche di lavoro	Avere collaborazione	Sostegno

4.3 Contributo

4.3.1 Attività predominanti svolte dagli sviluppatori con disabilità

RQ₂. Quali sono le attività maggiormente svolte dagli sviluppatori con disabilità?

Nonostante gli ostacoli che possono incontrare lungo il loro cammino, gli sviluppatori con disabilità dimostrano una straordinaria competenza nel portare a termine un’ampia gamma di attività predominanti nel campo dello sviluppo software.

Per meglio comprendere l'importanza del ruolo che svolgono, vengono presentati i dettagli di tali attività:

- **Sviluppo e programmazione software** [27, 3]: sono coinvolti nello sviluppo e nella programmazione software infatti, utilizzano le loro competenze tecniche per la scrittura di codice, la creazione di applicazioni e lo sviluppo di siti web. La ricerca ha, inoltre, evidenziato che trascorrono una parte significativa del loro tempo navigando nel codice per supportarne la comprensione e la manutenzione. A tal proposito, gli studi hanno dimostrato che un'efficace navigazione del codice è fondamentale per supportare, in questo caso gli sviluppatori con disabilità visive, nella costruzione di un modello cognitivo che possa facilitare la comprensione del flusso di codice e la posizione dei diversi blocchi. [27];
- **Revisione del codice** [3]: utilizzano le loro competenze tecniche per individuare bug, errori di programmazione e problemi di usabilità, contribuendo a migliorare la qualità del software;
- **Progettazione di interfacce utente** [29]: sono attivamente coinvolti nella progettazione di interfacce utente, tramite l'utilizzo di strumenti di prototipazione (quali, ad esempio, Sketch, Adobe XD, Balsamiq e UXPin) che possono essere utilizzati, nel contesto di disabilità visive, in combinazione con screen reader (quali, ad esempio, VoiceOver e Narrator);
- **Collaborazione e comunicazione** [3]: alcuni utilizzano la Pair Programming, cioè una pratica di sviluppo software in cui due programmatori lavorano fianco a fianco per scrivere e rivedere il codice. In tal caso, un programmatore è responsabile della digitazione del codice, mentre il collega fornisce istruzioni e feedback sul codice scritto. Ciò si traduce in maggiore qualità del codice, risoluzione creativa dei problemi, trasferimento di conoscenze tra i membri del team e maggiore soddisfazione lavorativa per i programmatori coinvolti [3]. Inoltre, spesso gli stessi programmatori devono modificare contemporaneamente lo stesso codice prestando attenzione ad aderire a regole concordate di scrittura e stile del codice per mantenere la leggibilità e la coerenza dello stesso. Questo

processo aiuta i team a trovare difetti nel codice, creare consapevolezza sul progetto e individuare soluzioni alternative ai problemi [3];

- **Consulenza e supporto tecnico** [29, 27, 3, 28]: utilizzano le loro competenze tecniche ed il loro bagaglio di esperienze personali per offrire consulenza e supporto tecnico ad aziende o individui che ne facciano richiesta con l’obiettivo di aiutare a ricavare approfondimenti sulle loro esperienze, risolvere problemi tecnici, implementare soluzioni accessibili o fornire assistenza nell’utilizzo di determinati strumenti o software. Possono lavorare a stretto contatto con ricercatori, tramite, ad esempio, la partecipazione a studi esplorativi o interviste, per analizzare strumenti o fornire soluzioni che soddisfino le proprie esigenze;
- **Ricerca** [28]: sono impiegati in contesti di ricerca quali, ad esempio, l’analisi dei dati o la scrittura accademica, al fine di contribuire alla scoperta di soluzioni innovative per superare le barriere tecnologiche e migliorare l’accessibilità.

Tabella 4.9: Paper che rispondono alla domanda di ricerca RQ₂

Domande di ricerca	Paper
RQ ₂	“Inclusive Multimodal Voice Interaction for Code Navigation” [27], “Usability, Accessibility and Social Entanglements in Advanced Tool Use by Vision Impaired Graduate Student” [28], “Accessibility of High-Fidelity Prototyping Tools” [29], “Understanding Accessibility and Collaboration in Programming for People with Visual Impairments” [3]

Tabella 4.10: Open Coding RQ₂

Codici	Concetti	Categoria
Navigazione del codice	Codifica	Attività
Compiti di ricerca	Ricerca	Attività
Fase di progettazione	Progettazione	Attività
Scrittura del codice	Codifica	Attività
Pianificazione del progetto	Pianificazione	Attività
Comunicazione con i colleghi del progetto	Comunicazione	Attività
Coordinamento con i colleghi per la scrittura del codice	Collaborazione	Attività
Scrittura del codice	Codifica	Attività
Revisione del codice	Revisione	Attività
Programmazione di coppia	Collaborazione	Attività
Progettazione software	Progettazione	Attività
Sviluppo interfaccia utente	Interfaccia utente	Attività

4.4 Sfida

4.4.1 Difficoltà affrontate dagli sviluppatori con disabilità

RQ_{3.1}. Quali difficoltà gli sviluppatori con disabilità si trovano ad affrontare?

Lo sviluppo di software è uno dei settori in più rapida crescita tuttavia, le persone con disabilità non sono ben rappresentate nel campo dell'informatica e dello sviluppo software [12]. Per tale motivo si focalizza l'attenzione sulle difficoltà che gli sviluppatori con disabilità affrontano, concentrandosi su coloro con disabilità visive e/o fisiche.

La sfida predominante affrontata dagli sviluppatori con disabilità è associata ad un'**accessibilità inadeguata** che impedisce di sfruttare appieno determinate tecnologie [12, 23, 27, 28, 3]. L'inaccessibilità, infatti, è definita come una disparità tra il modo in cui le persone con disabilità usano e gestiscono le tecnologie ed i servizi e il modo in cui le persone non disabili usano e gestiscono quelle stesse tecnologie e servizi [28]. Meno sviluppatori con disabilità entrano nel campo dello sviluppo software proprio a causa di problemi di accessibilità affrontati quando si tenta di apprendere e praticare la programmazione [24].

In “CodeTalk: Improving programming environment accessibility for visually impaired developers”, Potluri et al. hanno raggruppato le sfide di inaccessibilità affrontate dagli sviluppatori, nell’utilizzo di ambienti di programmazione, in quattro categorie [12]:

1. **Rilevabilità:** la capacità con cui un utente può trovare le caratteristiche del sistema. Ad esempio [12]:

- molte funzionalità degli IDE sono chiaramente visibili nell’interfaccia utente, ma sono nascoste all’interno di più livelli di gerarchia di navigazione per gli utenti che utilizzano screen reader;
- con ogni nuova versione dell’IDE, vengono aggiunte nuove funzionalità e le funzionalità esistenti vengono modificate. Molte di queste modifiche sono rappresentate visivamente e non esiste un approccio strutturato per gli sviluppatori con disabilità visive per essere informati delle stesse;

2. **Visibilità:** la maggior parte degli IDE per definizione utilizzano l’ampio spazio fornito dai monitor ad alta risoluzione per presentare molti aspetti del processo di sviluppo del programma in un’unica schermata. Gli IDE forniscono funzionalità che consentono agli sviluppatori vedenti di dare un senso alle informazioni rapidamente guardando lo schermo. L’input visivo, essendo un modo più attivo di acquisire informazioni, offre l’opportunità di riceverle senza interrompere l’attività corrente. Sfortunatamente, queste funzionalità non sono disponibili per gli sviluppatori con disabilità visive che spesso devono analizzare le informazioni in modo lineare. Ad esempio, i livelli di indentazione nei linguaggi di programmazione dipendenti dagli spazi bianchi, come Python, sono facili da percepire per gli sviluppatori vedenti, a differenza dei non vedenti che devono leggere il numero di spazi bianchi per ogni riga [12];

3. **Navigabilità del codice:**

- Disabilità visiva [23]: è stato dimostrato che i programmatori non vedenti hanno maggiori difficoltà a navigare nel codice rispetto alle loro controparti vedenti. Gli screen reader sono abili nel comunicare la struttura e le

convenzioni per navigare in strutture ad albero (ad esempio caselle postali, directory, ecc.) ma, sfortunatamente, non hanno accesso alla struttura di un programma. I token vengono letti uno alla volta e il programma è suddiviso in una serie di righe. La navigazione ne risente di conseguenza, con i programmatori costretti a usare i tasti freccia per leggere ogni riga di codice;

- **Disabilità fisica [27]:** la navigazione del codice può essere particolarmente sfidante per sviluppatori con disabilità fisiche a causa della dipendenza dall'uso di mouse e tastiera. Approcci comuni includono l'utilizzo di una funzione di ricerca locale che consente agli sviluppatori di trovare il codice in base alla corrispondenza lessicale, utilizzando un grafo delle chiamate in cui i metodi sono rappresentati come nodi che gli sviluppatori possono utilizzare per navigare tra diversi metodi. Molti IDE moderni forniscono anche un "trova tutti i riferimenti" che visualizza tutte le chiamate associate a un particolare metodo o funzione, nonché un "vai alla definizione", strumento che consente agli sviluppatori di trovare la definizione di un identificatore specificato. Tuttavia, questi approcci possono presentare dei limiti in quanto gli sviluppatori potrebbero dover passare dalla visualizzazione di un file all'altro e, quindi, ridimensionare o riorganizzare diversi riquadri di file, aumentando così i tempi di sviluppo ed il carico cognitivo nella comprensione del flusso di codice.

4. **Allertabilità:** gli IDE trasmettono una quantità significativa di informazioni in tempo reale attraverso un'interfaccia completamente visiva. Tali informazioni avvisano lo sviluppatore di problemi che richiedono attenzione immediata o azioni in corso. Ad esempio, le informazioni sugli errori di sintassi negli IDE sono fornite da segnali visivi come la linea tratteggiata rossa, che non sono disponibili per gli sviluppatori non vedenti in modo proattivo [12].

Da ciò se ne evince che i programmatori con disabilità, per completare attività specifiche, hanno necessità di utilizzare un complesso ecosistema di tecnologie [3] in quanto sono indotti a far uso di più strumenti per la risoluzione di compiti che in genere dovrebbero essere completati con un solo strumento [28]. Questo può essere

complesso per le persone con disabilità che potrebbero avere difficoltà a navigare tra i vari strumenti o ad utilizzarli contemporaneamente.

L'inaccessibilità delle tecnologie può comportare l'**esclusione** non solo dall'utilizzo di tali tecnologie ma anche dalle carriere tecniche [22, 25]. Un esempio è associato alle persone con disabilità fisiche che non sono in grado di utilizzare i tradizionali dispositivi di input (ad esempio mouse e tastiera) e che spesso sono escluse dalle professioni tecniche come lo sviluppo web [22].

Ciò che complica maggiormente l'accessibilità, qualora fosse garantita, è la problematica relata al **contesto collaborativo** [28, 3]. La maggior parte del software è costruito in collaborazione, infatti i programmatori spesso devono collaborare con gli altri membri del team, inclusi designer e project manager [3]. Alcuni sviluppatori con disabilità possono incontrare difficoltà a comunicare o interagire con gli altri complicando, in questo modo, la loro scelta degli strumenti, se i collaboratori avessero preferenze, richieste o sistemi specifici. Andrebbero ad adattare le preferenze dei collaboratori, ad esempio per fattori legati ad uno squilibrio di potere, alle proprie esigenze facendo sì che tali scelte ricadano e vadano a modificare l'accessibilità dello strumento o del processo [28].

Un'ulteriore area che fa emergere barriere per i programmatori con disabilità è individuabile nell'**istruzione** quindi, nella mancanza di un insegnamento appropriato nel campo dello sviluppo software [13]. Tali barriere permeano tutte le parti dell'istruzione, dall'accesso ai materiali, allo svolgimento dei compiti, al lavoro con i professori. In particolar modo, nel contesto della disabilità visiva, le sfide affrontate sono associate a [13]:

- Lezioni: manca il contesto poiché ciò di cui si parla non replica il contenuto visivo;
- Materiali: difficoltà ad ottenere libri di testo e dispense in un formato accessibile;
- Assegnazioni: incarichi sostitutivi o esoneri durante i corsi di informatica; questi esempi spaziano dall'uso di diverse tecnologie per completare i corsi, costruire un sistema diverso, sfruttare l'assistenza umana, ricevere voti in base ai componenti accessibili.

Un ostacolo chiave risiede principalmente nella **complessità dell'apprendimento di linguaggi e pratiche di programmazione** [24, 26, 4]. Di solito le persone con disabilità visive imparano a programmare tramite codifica basata su testo con l'aiuto di un mentore. Se vi è una lotta con questo tipo di ambienti, è probabile che il livello di motivazione diminuisca. Normalmente, per mantenere alta tale motivazione, si cerca di attirare l'attenzione tramite, ad esempio, le animazioni in Scratch [4]. Tuttavia, i linguaggi basati su blocchi sono dipendenti dalla grafica e, dunque, inaccessibili per persone con disabilità visive [26].

È possibile, in conclusione, racchiudere le sfide affrontate come segue:

Tabella 4.11: Sfide affrontate dagli sviluppatori con disabilità

#	Categoria	Sottocategorie
1	Accessibilità inadeguata	Rilevabilità, Visibilità, Navigabilità del codice, Alertabilità
2	Esclusione dalle carriere tecniche	
3	Difficoltà nel contesto collaborativo	
4	Istruzione come mancanza di un insegnamento appropriato	
5	Complessità dell'apprendimento di linguaggi e pratiche di programmazione	

Tabella 4.12: Paper che rispondono alla domanda di ricerca RQ_{3,1}

Domanda di ricerca	Paper
RQ _{3,1}	<p>“PSST: Enabling Blind or Visually Impaired Developers to Author Sonifications of Streaming Sensor Data” [22], “Tangible music programming blocks for visually impaired children” [4], “Accessible AST-based programming for visually-impaired programmers” [23], “Educational experiences of blind programmers” [13], “CodeTalk: Improving programming environment accessibility for visually impaired developers” [12], “Node-Read: A Visually Accessible Low-Code Software Development Extension” [24], “Voiceye: A Multimodal Inclusive Development Environment” [25], “StoryBlocks: A Tangible Programming Game To Create Accessible Audio Stories” [26], “Inclusive Multimodal Voice Interaction for Code Navigation” [27], “Usability, Accessibility and Social Entanglements in Advanced Tool Use by Vision Impaired Graduate Students” [28], “Understanding Accessibility and Collaboration in Programming for People with Visual Impairments” [3]</p>

Tabella 4.13: Open Coding RQ_{3,1}

Codici	Concetti	Categoria
Inaccessibilità delle tecnologie	Esclusione	Difficoltà
Mancanza degli strumenti	Complessità nell'apprendimento	Difficoltà
Difficoltà nella navigazione del codice	Inaccessibilità	Difficoltà
Difficoltà nella comprensione della struttura del codice	Inaccessibilità	Difficoltà
Barriere a partire dall'istruzione	Barriere nell'istruzione	Difficoltà
Accessibilità inadeguata	Inaccessibilità	Difficoltà
4 categorie di sfide di accessibilità	Sfide di accessibilità	Difficoltà
Difficoltà nell'apprendimento di linguaggi e pratiche di programmazione nuove	Complessità nell'apprendimento	Difficoltà
Esclusione dalle carriere tecniche a causa della predominanza di strumenti quali tastiera e mouse	Esclusione	Difficoltà
Inaccessibilità dei linguaggi basati su blocchi	Complessità nell'apprendimento	Difficoltà
Utilizzo intensivo di strumenti tradizionali inaccessibili	Inaccessibilità	Difficoltà
Utilizzo di più strumenti	Inaccessibilità	Difficoltà
Difficoltà a comunicare	Collaborazione	Difficoltà
Uso di più strumenti	Inaccessibilità	Difficoltà
Influenza nella collaborazione	Comunicazione	Difficoltà

4.4.2 Modi per affrontare le sfide degli sviluppatori con disabilità e attori responsabili di applicarle

RQ_{3,2}. Quali sono le strategie e i metodi per affrontare tali sfide? Chi deve applicarle?

Si analizzano, in primo luogo, le metodologie sviluppate per affrontare le sfide degli sviluppatori con disabilità, per poi concentrare l'attenzione su chi è responsabile di applicare queste strategie e metodi.

Vi sono diverse metodologie che possono essere utilizzate per affrontare le sfide degli sviluppatori con disabilità.

In particolare:

1. **Adattamenti tecnologici:** fornire adattamenti personalizzati per facilitare l'accesso e l'interazione con gli strumenti di sviluppo.

Per sviluppatori con disabilità visive si individuano:

- CodeTalk [12]: è un plug-in per Visual Studio che funziona con le versioni di Visual Studio 2015/2017 e supporta C# e Python ma, essendo sia modulare che estensibile, può essere facilmente implementato per altri IDE e persino altri linguaggi.

Aiuta gli sviluppatori che utilizzano lettori di schermo ad avere una chiara idea dei diversi costrutti di codice contenuti in un file tramite la funzionalità di riepilogo che fornisce una vista ad albero accessibile del codice. Inoltre, dal momento che gli sviluppatori interagiscono frequentemente con le funzioni in un file, per consentire una rapida navigabilità tra le stesse, CodeTalk consente una visualizzazione accessibile dell'elenco delle funzioni contenute nel file di codice corrente.

Sia il riepilogo del codice che l'elenco delle funzioni consentono la navigazione rapida del codice.

Fornisce, inoltre, informazioni sugli errori di sintassi in tempo reale tramite toni di errore proattivi. Lo sviluppatore può, quindi, premere una scorciatoia da tastiera per ottenere un elenco accessibile degli errori. Si propone, inoltre, un nuovo approccio al debugging: CodeTalk offre agli sviluppatori la possibilità di scegliere tra debugging vocale e non vocale, fornisce agli sviluppatori informazioni su variabili specifiche o valuta un'espressione nel contesto di esecuzione e offre un'opzione per interrompere o continuare l'esecuzione dopo il segnale audio;

- Node-Read [24]: è un'estensione per Node-RED (un software open source) che si concentra sul miglioramento della compatibilità con i lettori di schermo.

L'obiettivo principale dell'estensione si è basato sul mostrare all'utente le scorciatoie esistenti: si è scoperto che c'erano combinazioni di tasti comuni non erano intercettate dal momento che erano bloccate dagli screen reader.

Pertanto, alcune funzionalità di scorciatoia esistenti sono state tradotte in diverse combinazioni di scorciatoie per aggirare il problema dell'intercettazione. Sono state aggiunte, inoltre, anche scorciatoie aggiuntive per assistere nella navigazione dei nodi;

- CodeMirror-Blocks (CMB) [23]: è un toolkit per la creazione di ambienti di programmazione completamente accessibili. È progettato per essere estendibile ad altri linguaggi di programmazione infatti, quando viene fornito con un parser che soddisfa determinati requisiti CMB creerà un editor AST (Abstract Syntax Tree) completamente accessibile per quel linguaggio.

Inoltre, disaccoppia la sintassi testuale dall'etichetta descrittiva parlata per quel testo quindi, va a descrivere il codice invece di leggere la sintassi. Le descrizioni possono essere di natura pedagogica (ad esempio, "foo: una definizione di funzione che è pubblica, statica e produce un double") e possono essere personalizzate in base all'età o anche alla lingua parlata (ad esempio, "foo: una definición de función que es pública, estática y produce un doble").

Oltre a ciò, funziona interamente in un web browser in quanto CMB è costruito sulla popolare libreria CodeMirror; qualsiasi ambiente di programmazione che utilizza CodeMirror può essere abilitato all'accessibilità collegando questo strumento al parser appropriato.

Per sviluppatori con disabilità fisiche si individua:

- Interazione vocale multimodale per la navigazione del codice [27]: è stata effettuata la progettazione di un editor di codice che integra tre funzionalità di navigazione adattate per il controllo vocale multimodale cioè, "Trova tutti i riferimenti", "Trova/Cerca" e "Vai alla definizione".

In merito a "Trova tutti i riferimenti", tipicamente visualizza l'elenco di tutti i riferimenti rilevanti in relazione a un identificatore specificato dall'utente tramite input da tastiera (ad esempio una variabile, una funzione, ecc.). Per adattare questo approccio all'interazione vocale, gli utenti possono inizialmente posizionare il cursore (tramite i comandi vocali disponibili)

su un identificatore specifico all'interno del codice, poi emettere un comando per attivare il "Trova tutti i riferimenti" e un elenco di riferimenti viene visualizzato sul lato destro dell'interfaccia. Per navigare nell'elenco, ogni elemento è mappato con un valore numerico univoco (ad esempio, se "11" viene emesso come comando vocale, verrà selezionato l'undicesimo elemento).

"Trova/Cerca", invece, consente agli utenti di cercare il codice fornendo un termine di ricerca tramite la tastiera. Questa funzione è stata adattata per l'interazione vocale consentendo all'utente di emettere un comando verbale, "Trovare", per cercare il termine desiderato. Gli utenti possono, a questo punto, navigare tra i risultati della ricerca indicando il numero dell'istanza che desiderano selezionare.

Per "Vai alla definizione", questa consente agli sviluppatori di navigare direttamente alla definizione di un identificatore (ad esempio, una variabile, una funzione, ecc.). Per personalizzare questo per l'interazione vocale, l'utente può posizionare il cursore attorno a un identificatore (tramite i comandi vocali disponibili) e quindi emettere un comando per avviare tale funzionalità. Il cursore salta, quindi, alla relativa definizione dell'identificativo specifico scelto dall'utente e la navigazione è completata;

2. **Design inclusivo:** adottare un approccio di design inclusivo sin dall'inizio sia del processo di sviluppo del software che dell'insegnamento della programmazione.

Per sviluppatori con disabilità visive si individuano:

- Piattaforma musicale tangibile [4]: mira a insegnare le basi della programmazione attraverso la creazione di musica infatti, ordinando i blocchi tangibili in una struttura algoritmica è possibile creare una melodia.

Tali blocchi tangibili sono cartoncini tagliati al laser con adesivi NFC che devono essere inseriti in un riquadro cartaceo. Per ascoltare la funzionalità dei blocchi, per creare il codice e riprodurre la melodia è stata sviluppata un'applicazione Android.

Se si è in modalità informazioni, si può ascoltare la funzionalità dei blocchi. Se si è in modalità codice, è possibile creare il codice e riprodurre la melodia.

L'applicazione Android riconosce i blocchi con adesivi NFC mentre il codificatore sposta il telefono su ciascun blocco uno alla volta. Come ultimo passaggio, il blocco di riproduzione viene scansionato e la melodia viene riprodotta dagli altoparlanti;

- StoryBlocks [26]: è un gioco di programmazione tangibile per supportare i programmatori alle prime armi tramite la creazione di storie accessibili. Comprende una serie di blocchi tangibili, uno spazio di lavoro ampliato e un software per interpretare e riprodurre le storie dei programmatori.

In merito ai blocchi, ognuno rappresenta un componente di una storia interattiva ed è differenziabile utilizzando solo informazioni tattili tangibili in modo tale che: i personaggi vengano definiti da blocchi con bordi quadrati, le azioni da blocchi con bordi arrotondati e le dichiarazioni di controllo da blocchi con bordi a forma di diamante.

Le storie vengono create in un'area di lavoro designata e, una volta completate, le immagini vengono catturate da una webcam dall'alto e interpretate da un programma Python personalizzato.

Il sistema StoryBlocks legge ogni blocco da sinistra a destra, ordina i blocchi e genera una parola associata a ciascun blocco. Combina, quindi, tutte le parole in una frase e aggiunge voci diverse ai personaggi ed effetti sonori alle azioni. Ad esempio, i blocchi Gatto → Corsa → Topo potrebbe essere una frase letta come "il gatto corre dietro al topo" con la voce acuta del gatto e l'aggiunta di un suono di un personaggio dei cartoni che corre.

Per sviluppatori con disabilità fisiche si individua:

- Voiceye [25]: è una nuova applicazione multimodale realizzata per la scrittura e la modifica del codice HTML e CSS. È costituita da quattro componenti: un riconoscimento vocale automatico (che utilizza il servizio Azure Speech di Microsoft per comandi verbali come la selezione,

la navigazione e la rimozione del codice, nonché per dettare forme più lunghe di testo come i commenti), una grammatica della sintassi basata su regole, una tastiera su schermo (controllata tramite l'input dello sguardo) e due interruttori meccanici di cui uno per selezionare i pulsanti all'interno l'interfaccia (ad esempio un pulsante della tastiera virtuale), mentre l'altro per attivare/ disattivare il sistema di riconoscimento vocale. Mentre l'utente parla, il prototipo converte la voce in testo ed esegue un controllo rispetto alla grammatica della sintassi basata su regole definite. Le azioni appropriate vengono, quindi, attivate in base al comando vocale. La tastiera su schermo consente agli utenti di digitare il codice fissandosi sui caratteri di interesse e, quindi, completando la selezione premendo uno degli interruttori meccanici designati.

Viene mostrato di seguito un esempio:

Tabella 4.14: Esempio applicazione Voiceye

Compito	Comando vocale	Descrizione	Esempio di codice/Espressione	di	Esempio di output
"Aggiungere commento"	"Commento"	Aggiungi commento riga singola	padding: 4px 2px 3px 4px; parola "Commento"	/*padding: 4px 2px 3px 4px;*/	

Riassumendo, dunque, gli strumenti che supportano le persone con disabilità nello sviluppo del software, categorizzati per tipologie di disabilità, sono:

- Disabilità visive: CodeTalk, Node-Read, CodeMirror-Blocks, Piattaforma musicale tangibile, StoryBlocks;
- Disabilità fisiche: Interazione vocale multimodale per la navigazione del codice, Voiceye.

3. **Formazione** [13]: per garantire un'istruzione accessibile a tutti, è fondamentale fornire materiali didattici adattati alle disabilità del singolo. Nel contesto della disabilità visiva la creazione di formati tattili delle immagini contenute in un libro di testo è un passo importante per consentire di accedere alle informazioni visive. Tactile Graphics Assistant (TGA)⁵ è uno dei programmi più utilizzati

⁵<https://tactilegraphics.cs.washington.edu/>

a tal scopo ed offre strumenti e risorse per semplificare e automatizzare la creazione di formati tattili, consentendo ai docenti di creare materiali accessibili in modo più efficiente. Relativamente ai docenti è importante che ricevano un adeguato supporto per creare un ambiente di apprendimento inclusivo, fornendo loro programmi di formazione specifici sull'insegnamento alle persone con disabilità, risorse appropriate e assistenza di professionisti dell'istruzione specializzati nel campo della disabilità;

4. **Collaborazione** [28, 3]: le comunità di ingegneria del software, CSCW (Computer-Supported Cooperative Work) e HCI (Human-Computer Interaction) hanno riconosciuto l'importanza della collaborazione e della comunicazione nella programmazione. Per coordinare lo sviluppo e la manutenzione di software complessi, il processo inizia spesso con la pianificazione dell'architettura software che comunica le relazioni tra diversi componenti come il database, i server e l'interfaccia utente. Ciò consente ai membri del team di sviluppare un vocabolario comune e facilitare la comunicazione [3]. Inoltre, fondamentale è cercare e fornire assistenza: bisogna tentare di avere fiducia che coloro che prestano aiuto non giudicheranno per aver cercato assistenza e, al tempo stesso, assistere i colleghi con i loro problemi dà l'opportunità di mitigare le preoccupazioni sulla ricerca di aiuto [28, 3]. Dunque, la collaborazione tra gli sviluppatori con disabilità e i loro colleghi è utile per condividere conoscenze e affrontare le sfide in modo collettivo.

È importante che queste strategie siano applicate in modo collaborativo e continuo, al fine di garantire un ambiente di lavoro equo e inclusivo per gli sviluppatori con disabilità.

La responsabilità di applicare queste strategie e metodi spetta a diversi attori:

- **Le aziende, i datori di lavoro e il personale educativo** devono fornire un ambiente di lavoro inclusivo e risorse adeguate per consentire agli sviluppatori con disabilità di svolgere il proprio lavoro in modo efficace [13];
- **Gli sviluppatori** stessi possono collaborare con i loro colleghi per superare le sfide specifiche che incontrano [28, 3];

- **Le comunità di sviluppatori e le organizzazioni del settore** possono promuovere la sensibilizzazione, condividere risorse e best practice e lavorare per creare un settore tecnologico più inclusivo e accessibile [4, 22].

Tabella 4.15: Paper che rispondono alla domanda di ricerca RQ_{3.2}

Domanda di ricerca	Paper
RQ _{3.2}	<p>“PSST: Enabling Blind or Visually Impaired Developers to Author Sonifications of Streaming Sensor Data” [22], “Tangible music programming blocks for visually impaired children” [4], “Accessible AST-based programming for visually-impaired programmers” [23], “CodeTalk: Improving programming environment accessibility for visually impaired developers” [12], “Node-Read: A Visually Accessible Low-Code Software Development Extension” [24], “Voiceye: A Multimodal Inclusive Development Environment” [25], “StoryBlocks: A Tangible Programming Game To Create Accessible Audio Stories” [26], “Inclusive Multimodal Voice Interaction for Code Navigation” [27], “Educational Experiences of Blind Programmers Usability” [13], “Accessibility and Social Entanglements in Advanced Tool Use by Vision Impaired Graduate Students” [28], “Understanding Accessibility and Collaboration in Programming for People with Visual Impairments” [3]</p>

Tabella 4.16: Open Coding RQ_{3.2}

Codici	Concetti	Categoria
Ppst	Design inclusivo	Tipologia
Piattaforma musicale tangibile	Design inclusivo	Tipologia
CodeMirror-Blocks	Adattamenti tecnologici	Tipologia
Codetalk	Adattamenti tecnologici	Tipologia
Node Read	Adattamenti tecnologici	Tipologia
Voiceye	Design inclusivo	Tipologia
Storybloks	Design inclusivo	Tipologia
Interazione Vocale Multimodale	Adattamenti tecnologici	Tipologia
Miglioramenti da attuare nei contesti di apprendimento	Formazione	Tipologia
Modifiche nei contesti collaborativi	Collaborazione	Tipologia
Miglioramenti nei contesti collaborativi	Collaborazione	Tipologia

CAPITOLO 5

Conclusioni

L'obiettivo di questo capitolo è presentare una sintesi del lavoro svolto e valutare i risultati ottenuti.

5.1 Analisi e sintesi dei risultati

L'obiettivo della ricerca è stato effettuare una systematic literature review per fornire una panoramica che mirasse a valutare lo stato dell'arte attuale rispetto ai problemi che gli sviluppatori con disabilità affrontano sul lavoro così da ottenere possibili soluzioni adatte a supportare tali sviluppatori.

I risultati lasciano emergere che ciò che motiva le persone con disabilità a lavorare come sviluppatori software è dettato dal ricevere un costante incoraggiamento ad acquisire competenze di programmazione e dal ricevere un supporto appropriato, anche in termini di accesso a strumenti tecnologici adeguati, che consenta di eliminare le barriere che potrebbero altrimenti limitare la partecipazione delle persone con disabilità nel settore dello sviluppo software. In effetti, si è dimostrato che avere esperienze positive può incoraggiare ed entusiasmare le persone con disabilità a lavorare come sviluppatori software.

La partecipazione delle persone con disabilità nello sviluppo software richiede il sostegno e l'incentivazione di diverse parti interessate. Questi attori chiave includono il personale educativo che promuove l'inclusione scolastica degli studenti con disabilità e la formazione degli insegnanti affinché possano comprendere le esigenze specifiche degli studenti con disabilità, la comunità di sviluppatori che fornisce un supporto funzionale nel contesto lavorativo così da facilitare la partecipazione delle persone con disabilità, le organizzazioni di supporto che sensibilizzano e offrono sostegno nel settore dello sviluppo software e le aziende del settore tecnologico.

Affinché si possa sostenere la partecipazione delle persone con disabilità al mondo dello sviluppo software è necessario condurre ricerche approfondite per identificare i bisogni specifici delle persone con disabilità, progettare soluzioni inclusive e sviluppare tecnologie accessibili. È fondamentale migliorare costantemente le tecnologie utilizzate per fornire un migliore supporto alle persone con disabilità. Inoltre, creare opportunità di apprendimento e formazione, come corsi accessibili e inclusivi, è essenziale per coloro che desiderano entrare nel campo dello sviluppo software, così come promuovere l'interazione sociale e la collaborazione per favorire la partecipazione attiva da parte di tutti.

Nonostante gli ostacoli che gli sviluppatori con disabilità possono incontrare, questi svolgono un ruolo importante nel campo dello sviluppo software. Le loro attività includono lo sviluppo e la programmazione software, la revisione del codice, la progettazione di interfacce utente, la collaborazione e la comunicazione tra colleghi, la consulenza e il supporto tecnico ad aziende o individui che ne facciano richiesta, e la partecipazione alle ricerche per superare le barriere tecnologiche e migliorare l'accessibilità.

Gli ostacoli citati sono in gran parte legati all'accessibilità inadeguata che riguarda, prevalentemente, la rilevabilità, la visibilità, la navigabilità del codice e l'allertabilità. Tali sfide portano gli sviluppatori con disabilità ad utilizzare un complesso ecosistema di tecnologie in quanto sono indotti a far uso di più strumenti per la risoluzione di compiti che in genere dovrebbero essere completati con un solo strumento e, inoltre, ciò può contribuire all'esclusione dalle carriere tecniche. L'accessibilità può essere messa a repentaglio dal contesto collaborativo dal momento che alcuni sviluppatori con disabilità possono incontrare difficoltà a comunicare o interagire con gli altri.

Ostacoli sono affrontati anche nell'istruzione quindi, nella mancanza di un insegnamento appropriato nel campo dello sviluppo software così come nell'apprendimento di linguaggi e pratiche di programmazione.

Le metodologie atte ad affrontare tali sfide includono adattamenti tecnologici come CodeTalk, Node-Read e CodeMirror-Blocks, che facilitano l'accesso e l'interazione con gli strumenti di sviluppo. Il design inclusivo è un approccio fondamentale, sin dall'inizio del processo di sviluppo del software e dell'insegnamento della programmazione. In merito a quest'ultimo aspetto, è fondamentale fornire materiali didattici adattati alle disabilità del singolo. Inoltre, la collaborazione tra gli sviluppatori con disabilità e i loro colleghi è essenziale per condividere conoscenze e affrontare le sfide in modo collettivo. La responsabilità di applicare queste strategie e metodi spetta alle aziende, ai datori di lavoro, al personale educativo, alle comunità di sviluppatori e alle organizzazioni del settore tecnologico.

La presente ricerca potrebbe avere un impatto significativo nel campo dell'inclusione e dell'occupazione delle persone con disabilità nel settore dello sviluppo software.

Attraverso l'analisi delle ragioni per cui le persone con disabilità scelgono di lavorare come sviluppatori software, si potrebbe ottenere una migliore comprensione delle motivazioni che spingono queste persone a intraprendere questa carriera.

Inoltre, identificare gli attori chiave responsabili di promuovere la partecipazione delle persone con disabilità nel mondo dello sviluppo software può facilitare la creazione di collaborazioni tra parti interessate a favorire l'inclusione.

Le azioni ed iniziative proposte per sostenere la partecipazione delle persone con disabilità possono fungere da guida per l'implementazione di politiche e programmi mirati a rimuovere le barriere e adattare l'ambiente di lavoro per favorire l'accessibilità e l'equità.

L'analisi delle attività predominanti svolte dagli sviluppatori con disabilità può fornire un quadro completo delle competenze e delle capacità che queste persone portano nel campo dello sviluppo software, dimostrando che le disabilità non sono un ostacolo per svolgere con successo tali ruoli.

Conoscere le difficoltà affrontate dagli sviluppatori con disabilità può aiutare a sviluppare strategie e soluzioni per superare tali ostacoli.

Infine, identificare i modi per affrontare le sfide degli sviluppatori con disabilità e gli attori responsabili di applicarle può contribuire a creare un ambiente di lavoro più inclusivo, fornendo supporto, adattamenti e risorse necessarie per garantire il successo e il benessere degli sviluppatori con disabilità.

In sintesi, la ricerca proposta ha il potenziale per generare conoscenze preziose e informazioni pratiche che potrebbero influenzare positivamente l'occupazione e l'inclusione delle persone con disabilità nel settore dello sviluppo software.

Bibliografia

- [1] K. Albusays, P. Bjorn, L. Dabbish, D. Ford, E. Murphy-Hill, A. Serebrenik, and M.-A. Storey, "The diversity crisis in software development," *IEEE Software*, vol. 38, no. 2, pp. 19–25, 2021. (Citato alle pagine 1 e 6)
- [2] M. D. Salute, "Relazione del ministro della salute sullo stato di attuazione delle politiche inerenti la prevenzione della cecità, l'educazione e la riabilitazione visiva," 2023. (Citato a pagina 2)
- [3] M. Pandey, V. Kameswaran, H. V. Rao, S. O'Modhrain, and S. Oney, "Understanding accessibility and collaboration in programming for people with visual impairments," *Proc. ACM Hum.-Comput. Interact.*, vol. 5, 2021. (Citato alle pagine 2, 29, 34, 35, 36, 37, 38, 40, 41, 43, 50 e 51)
- [4] A. Sabuncuoglu, "Tangible music programming blocks for visually impaired children," pp. 423–429, 2020. (Citato alle pagine 2, 29, 30, 31, 32, 33, 34, 35, 42, 43, 47 e 51)
- [5] G. Castaldo, "Online appendix," 2023. [Online]. Available: <https://docs.google.com/spreadsheets/d/1z5nRkqr9-9G60iWRoL9g3FekH7w52gR1/edit?usp=sharing&ouid=116815586286109557480&rtpof=true&sd=true> (Citato alle pagine 4, 27, 28 e 29)

-
- [6] G. Rodriguez, R. Nadri, and M. Nagappan, "Perceived diversity in software engineering: a systematic literature review," *Empirical Software Engineering*, vol. 26, 2021. (Citato alle pagine 6 e 7)
- [7] D. E. Byrne, *The Attraction Paradigm*. Academic Press, 1971. (Citato a pagina 6)
- [8] J. C. Turner, *Rediscovering the Social Group: A Self-categorization Theory*. Basil Blackwell, 1987. (Citato a pagina 7)
- [9] A. Homan, D. Knippenberg, G. van Kleef, and C. De Dreu, "Bridging faultlines by valuing diversity: Diversity beliefs, information elaboration, and performance in diverse work groups," *Journal of Applied Psychology*, vol. 92, p. 1189–1199, 2007. (Citato a pagina 7)
- [10] U. Nations, "Convention on the rights of persons with disabilities," 2017. (Citato a pagina 7)
- [11] M. Adesina, I. Olufadewa, R. Oladele, I. Oduguwa, and F. Abudu, "Historical perspective and classification of disability," vol. 3, pp. 17–24, 2021. (Citato alle pagine 7 e 9)
- [12] V. Potluri, P. Vaithilingam, S. Iyengar, Y. Vidhya, M. Swaminathan, and G. Srinivasa, "Codetalk: Improving programming environment accessibility for visually impaired developers," vol. 2018-April, 2018. (Citato alle pagine 9, 29, 33, 34, 35, 38, 39, 40, 43, 45 e 51)
- [13] C. Baker, C. Bennett, and R. Ladner, "Educational experiences of blind programmers," pp. 759–765, 2019. (Citato alle pagine 9, 14, 29, 30, 31, 32, 33, 34, 35, 41, 43, 49, 50 e 51)
- [14] D. M. B. Paiva, A. P. Freire, and R. P. de Mattos Fortes, "Accessibility and software engineering processes: A systematic literature review," *Journal of Systems and Software*, vol. 171, 2021. (Citato alle pagine 11 e 13)
- [15] Menezes and R. Prikladnicki, "Diversity in software engineering," 05 2018, pp. 45–48. (Citato a pagina 11)

-
- [16] A. Mountapmbeme, O. Okafor, and S. Ludi, "Addressing accessibility barriers in programming for people with visual impairments: A literature review," *ACM Trans. Access. Comput.*, vol. 15, no. 1, mar 2022. (Citato a pagina 12)
- [17] K. Naik, P. Tripathy, and a. O. M. C. Safari, *Software Evolution and Maintenance*. John Wiley & Sons, 2014. (Citato alle pagine 15, 16 e 17)
- [18] G. Lacerda, F. Petrillo, M. Pimenta, and Y.-G. Guéhéneuc, "Code smells and refactoring: A tertiary systematic review of challenges and observations," *Journal of Systems and Software*, vol. 167, p. 110610, 2020. (Citato alle pagine 15, 16 e 17)
- [19] S. Keele *et al.*, "Guidelines for performing systematic literature reviews in software engineering," 2007. (Citato alle pagine 18 e 21)
- [20] R. Hoda and J. Noble, "Becoming agile: A grounded theory of agile transitions in practice," pp. 141–151, 2017. (Citato alle pagine 23 e 24)
- [21] J. Wolfswinkel, E. Furtmueller, and C. Wilderom, "Using grounded theory as a method for rigorously reviewing literature," *European Journal of Information Systems*, vol. 22, 2013. (Citato a pagina 24)
- [22] V. Potluri, J. Thompson, J. Devine, B. Lee, N. Morsi, P. De Halleux, S. Hodges, and J. Mankoff, "Psst: Enabling blind or visually impaired developers to author sonifications of streaming sensor data," 2022. (Citato alle pagine 29, 41, 43 e 51)
- [23] E. Schanzer, S. Bahram, and S. Krishnamurthi, "Accessible ast-based programming for visually-impaired programmers," pp. 773–779, 2019. (Citato alle pagine 29, 38, 39, 43, 46 e 51)
- [24] L. Anderson, B. Barker, A. Reid, K. Lin, H. Khalajzadeh, and J. Grundy, "Node-read: A visually accessible low-code software development extension," p. 808–815, 2022. (Citato alle pagine 29, 38, 42, 43, 45 e 51)
- [25] B. Paudyal, C. Creed, M. Frutos-Pascual, and I. Williams, "Voiceeye: A multimodal inclusive development environment," p. 21–33, 2020. (Citato alle pagine 29, 34, 35, 41, 43, 48 e 51)

- [26] V. Koushik, D. Guinness, and S. K. Kane, “Storyblocks: A tangible programming game to create accessible audio stories,” p. 1–12, 2019. (Citato alle pagine 29, 32, 33, 42, 43, 48 e 51)
- [27] B. Paudyal, C. Creed, I. Williams, and M. Frutos-Pascual, “Inclusive multimodal voice interaction for code navigation,” p. 509–519, 2022. (Citato alle pagine 29, 36, 37, 38, 40, 43, 46 e 51)
- [28] K. Shinohara, M. Tamjeed, M. McQuaid, and D. A. Barkins, “Usability, accessibility and social entanglements in advanced tool use by vision impaired graduate students,” *Proc. ACM Hum.-Comput. Interact.*, vol. 6, 2022. (Citato alle pagine 29, 32, 33, 37, 38, 40, 41, 43, 50 e 51)
- [29] J. Li, G. W. Tigwell, and K. Shinohara, “Accessibility of high-fidelity prototyping tools,” 2021. (Citato alle pagine 29, 36 e 37)

Ringraziamenti

Mi è doveroso dedicare questo spazio del mio elaborato alle persone che hanno contribuito, con il loro instancabile supporto, alla realizzazione dello stesso.

Vorrei iniziare ringraziando la mia Relatrice, la Prof.ssa Filomena Ferrucci, i cui insegnamenti sono stati e saranno sempre di ispirazione per il mio lavoro. Mi ha fornito un bagaglio importante, sia culturale che di impostazione mentale, che porterò ovunque con me.

Ringrazio, inoltre, i Dottorandi Giulia Sellitto e Stefano Lambiase per avermi dato l'occasione di collaborare insieme, per l'immensa pazienza e gli indispensabili consigli, nonché per le conoscenze trasmesse durante tutto il percorso.

Questa tesi ha contribuito a piantare un albero in Colombia tramite il progetto Treedom.

<https://www.treedom.net/it/user/sesalab/event/sesa-random-forest>