

## TINJAUAN SISTEMATIS TERHADAP PERSYARATAN MANAJEMEN PERUBAHAN DALAM PROYEK PENGEMBANGAN PERANGKAT LUNAK

Anita Safitri

Universitas Negeri Surabaya

### Abstract

*Changes in software development are unavoidable and can occur due to changing needs of users, stakeholders, and the presence of new technologies. Analysis of the impact of changes greatly affects the cost and time of project work that can affect the success of the project. With the analysis, can help reduce the impact caused. Therefore, proper change management in software development projects is required. In this paper we present a review of the need for change management in software development. The writing of this paper is done to answer the problems that arise based on the discussion of the journals studied. The problems include the causes of change management, the impact of change, the methods can be used to cope with change, and tools can be used to cope with change. The method used in writing this review is to draw conclusions based on the discussion obtained from the grouping of journals related to change management and the need for change management sourced from ScienceDirect and IEEE. The results obtained from this review are to answer four questions that arise and draw conclusions about the methods and tools that are often used in large case studies in change management. The method that is often used to support the management of requirements change is Requirements Change Management (RCM). RCM consists of 3 stages (1) Understanding changes (2) Analysis of changes and (3) Finalizing changes to this method can be assisted with web tools. The web is often used to support change management needs because it can be used to manage quantitative and qualitative data.*

**Keywords:** Change Management; Change Management; Requirements Change Management

Received: 11 Mei 2022; Accepted: 20 Juni 2022; Published: 29 Juli 2022

\*Corresponding author

Email: [anitasafitri@unesa.ac.id](mailto:anitasafitri@unesa.ac.id)

To cite this document:

Safitri, Anita. *JDBIM (Journal of Digital Business and Innovation Management)*, Vol.1, No.1, Juni 2022, pp.69-78.

### **Abstrak**

*Perubahan dalam pengembangan perangkat lunak tidak dapat dihindari dan dapat terjadi karena perubahan kebutuhan pengguna, pemangku kepentingan, dan adanya teknologi baru. Analisis dampak dari perubahan sangat berpengaruh terhadap biaya dan waktu pengerjaan proyek yang dapat mempengaruhi keberhasilan proyek. Dengan adanya analisis, dapat membantu mengurangi dampak yang diakibatkan. Oleh karena itu diperlukan manajemen perubahan yang tepat dalam proyek pengembangan perangkat lunak. Dalam paper ini kami menyajikan review mengenai kebutuhan manajemen perubahan dalam pengembangan perangkat lunak. Adapun penulisan paper ini dilakukan untuk menjawab permasalahan yang muncul berdasarkan dari pembahasan dari jurnal yang dipelajari. Adapun permasalahan tersebut antara lain : 1. apa saja penyebab terjadinya manajemen perubahan, 2. Bagaimana dampak dari perubahan? 3. Apa metode yang dapat digunakan untuk mengatasi perubahan? 4. Apa alat bantu yang dapat digunakan untuk mengatasi perubahan? Metode yang digunakan dalam penulisan review ini adalah dengan menarik kesimpulan berdasarkan pembahasan yang didapat dari pengelompokan jurnal terkait dengan manajemen perubahan dan kebutuhan akan manajemen perubahan yang bersumber dari ScienceDirect dan IEEE. Hasil yang didapat dari review ini adalah untuk menjawab empat pertanyaan yang muncul dan menarik kesimpulan mengenai metode dan alat bantu yang sering digunakan dalam studi kasus besar dalam manajemen perubahan. Metode yang sering digunakan dalam mendukung manajemen perubahan kebutuhan adalah Requirement Change Management (RCM). RCM terdiri dari 3 tahapan (1) Memahami perubahan (2) Analisis perubahan dan (3) Finalisasi perubahan metode ini dapat dibantu dengan alat bantu web. Web sering digunakan dalam mendukung manajemen perubahan kebutuhan karena dapat digunakan untuk mengelola data kuantitatif dan kualitatif.*

*Kata Kunci : Manajemen Perubahan; Change Management; Requirement Change Management*

### **PENDAHULUAN**

Seiring dengan berjalannya waktu dan teknologi yang semakin berkembang, industri pengembangan perangkat lunak kian meningkatkan eksistensinya dalam bidang teknologi terbaru. Sistem perangkat lunak yang semakin kompleks mengikuti kebutuhan bisnis saat ini. Dalam pengembangan perangkat lunak, kebutuhan-kebutuhan dari pengembangan perangkat lunak didapatkan dari pengguna dan para pemangku kepentingan melalui berbagai teknik elisitasi. Dimana kebutuhan-kebutuhan tersebut merupakan suatu tujuan yang harus dicapai oleh pengembang perangkat lunak.

Dalam kebutuhan pengguna, pengguna sering kali menginginkan perubahan pada sistem yang akan dibuat. Perubahan kebutuhan itu dapat terjadi selama siklus pengembangan perangkat lunak atau setelah para pengembang perangkat lunak berasumsi bahwa kebutuhan perangkat lunak sudah dalam tahap finalisasi. Perubahan itu seringkali terjadi, dikarenakan terjadinya celah pada tahap elisitasi yang dilakukan oleh pengembang perangkat lunak, atau terbentur oleh regulasi perusahaan atau pemerintahan.

Tak jarang dalam memenuhi kebutuhan perubahan membuat biaya yang semakin mahal dan memakan waktu yang lebih lama. Untuk mengurangi biaya dan waktu akibat adanya kebutuhan perubahan, dibutuhkan manajemen perubahan yang sedini mungkin diterapkan dan diharapkan berhasil dalam mengembangkan sistem perangkat lunak. Manajemen kebutuhan ini tidak hanya penting, namun bersifat kritis yang dapat berdampak kepada keberhasilan proyek. Pada saat kebutuhan perubahan diajukan oleh pengguna, akan terdapat dampak dari adanya perubahan kebutuhan. Perlu adanya analisis pada bagian mana dari perangkat lunak yang akan mengalami perubahan. Dari hasil analisis tersebut dapat disimpulkan dampak apa saja yang bisa terjadi karena adanya kebutuhan perubahan. Analisis dampak dapat menggunakan metode *formal semantics* dengan tiga aktifitas untuk analisis dampak. Yang pertama, ahli dari kebutuhan perubahan mengusulkan perubahan dengan klasifikasi yang ada sebelumnya sebelum dilakukan perubahan yang sebenarnya. Kemudian yang kedua ahli harus mengidentifikasi apa saja yang akan mendapat dampak dari perubahan. dan yang ketiga yaitu, mengidentifikasi perubahan – perubahan apa saja yang mungkin terjadi (Goknil et al., 2014).

Selain itu, terdapat metode Conditional Random Field (CRF) merupakan metode yang digunakan untuk analisis dampak dari perubahan dengan dasar matematika yang mengusulkan untuk mengeksplorasi kebutuhan secara kuantitatif. CRF adalah metode statistik yang digunakan untuk encode hubungan pengamatan dan interpretasi yang konsisten (Xie, Yang, Chang, dan Liu, 2017) atau untuk menganalisa dampak dapat menggunakan metode Change Impact Analysis yang terdiri dari dua proses utama, (1) Memberikan perubahan dalam fitur (menambahkan menghapus atau updating), dan (2) memberikan perubahan dalam suatu arsitektur kerja yang menyadari perubahan dalam fitur (Díaz, Pérez, Garbajosa, dan Yagüe, 2013).

Dalam manajemen perubahan kebutuhan, pengembang perlu untuk melakukan komunikasi baik secara formal maupun informal dengan para pemangku kepentingan. Dalam proses manajemen perubahan terdapat beberapa metode yang dapat dilakukan, salah satunya adalah dengan metode *Requirement Change Management* (RCM). RCM terdiri dari tiga tahapan: (1) Memahami perubahan, (2) Analisa perubahan, dan (3) Finalisasi terhadap perubahan kebutuhan (Ali dan Lai, 2016).

Proses manajemen perubahan juga dapat dilakukan dengan menggunakan metode *Features Tree*. Metode ini dilakukan dalam dua tahapan utama, (1) membangun bentuk awal dari *Features Tree* dan (2) manajemen perubahan kebutuhan. Bentuk awal dari *Features Tree* berisikan gambaran kebutuhan yang mungkin didapatkan dari pengembangan perangkat lunak. Kemudian tim pengembang melakukan manajemen perubahan kebutuhan dengan cara melakukan persetujuan terhadap perubahan yang baru atau tetap menggunakan kebutuhan yang sudah teridentifikasi sebelumnya. Adapun metode ini dapat dilakukan dengan tiga aktifitas: (1) modifikasi terhadap perubahan dengan cara menambah, menghapus,

atau memperbarui, (2) mengontrol perubahan dengan cari menyetujui perubahan, atau tetap dengan kebutuhan yang sudah ada sebelumnya, dan (3) mencari hubungan ketergantungan antar kebutuhan dalam hal implikasi dan pengecualian (Lloyd, Moawad, dan Kadry, 2017).

Ada beberapa metode lain yang dapat digunakan salah satunya adalah metode *Partial Least Square* (PLS). PLS digunakan dalam analisis bisnis karena beberapa alasan (1) penelitian tentang hubungan kebutuhan yang relatif baru, dan (2) tidak ada model pengukuran yang sudah tersedia. PLS bisa menjadi teknik yang cocok untuk digunakan ketika fenomena yang akan diteliti masih relatif baru (Aziz dan Wang, 2010).

Dalam mendukung metode manajemen perubahan dibutuhkan tools atau alat bantu yang dapat membantu penerapan metode yang digunakan. Alat bantu berkonsentrasi pada pengelolaan kebutuhan perubahan dalam pengembangan perangkat lunak. Ada beberapa alat bantu yang dapat digunakan. Salah satunya adalah alat bantu yang berbasis web yang dimaksudkan untuk memfasilitasi dan mengelola pemodelan kebutuhan dan kebutuhan perubahan dalam semua fase. Dimulai dari kebutuhan pada saat fase inisiasi proyek dan tetap dapat melacak kebutuhan dan perubahan selama semua fase (Lloyd et al., 2017).

Alat bantu yang lain adalah dengan menggunakan Tool *Requirements Inferencing and Consistency Checking* (TRIC). TRIC merupakan alat dengan fitur untuk analisis dampak perubahan kebutuhan. alat ini dapat di unduh pada (<http://trese.cs.utwente.nl/tric/>) (Goknill, Kurtev, van der Berg, dan Spijkerman, 2014). Selain itu untuk mendukung metode *Conditional Random Field* (CRF) dibutuhkan alat bantu CRF++ *open source* yang menyediakan template fitur secara otomatis dengan format tertentu (Xie et al., 2017). *Review* ini mengacu pada literatur review sebelumnya yaitu *A systematic review of requirements change management* (Jayatilleke dan Lai, 2018) yang menggunakan metode *review* sistematis untuk mengidentifikasi kelebihan dan kekurangan dari teknik *Requirement Change Management* (RCM).

### **Pertanyaan Penelitian**

Untuk mendapatkan pemahaman tentang perubahan dalam pengembangan perangkat lunak, kami memformulasikan empat pertanyaan berikut :

1. Apa saja hal – hal yang dapat menyebabkan manajemen perubahan?  
Motivasi dibalik pertanyaan ini adalah untuk memahami apa saja hal – hal yang dapat menyebabkan adanya manajemen perubahan. untuk menjawab pertanyaan ini, kami menggunakan studi literature dari jurnal dengan beberapa kasus yang berbeda.
2. Bagaimana dampak dari perubahan?

Motivasi dibalik pertanyaan ini adalah untuk memahami apa saja akibat dari adanya kebutuhan perubahan yang terjadi dalam pengembangan perangkat lunak. Untuk menjawab pertanyaan ini, dengan studi literatur jurnal yang membahas tentang dampak dari adanya kebutuhan perubahan.

3. Apa metode yang digunakan untuk mengatasi kebutuhan perubahan?

Motivasi dibalik pertanyaan ketiga adalah untuk memahami metode apa saja yang dapat digunakan untuk mengatasi kebutuhan perubahan dan bisa diterapkan dalam manajemen perubahan.

4. Apa alat bantu yang digunakan untuk mengatasi perubahan?

Motivasi dibalik pertanyaan terakhir adalah untuk memahami alat bantu apa saja yang dapat mendukung manajemen perubahan dalam mengatasi kebutuhan perubahan.

## METODE

### Pendekatan Review

Pembuatan paper ini dilakukan dalam beberapa tahapan pendekatan : (1) Menentukan Topik, (2) mencari jurnal berdasarkan kata kunci, (3) filterasi jurnal berdasarkan beberapa pertanyaan penting, (4) pemberian bobot (5) melakukan klasifikasi jurnal, dan (6) menarik kesimpulan.



**Gambar 1. Tahapan Pendekatan Review**

1. Menentukan Topik

Pada tahap awal, kami menentukan topik yang akan kami angkat, dan dalam penulisan ini kami mengangkat topik tentang manajemen terhadap perubahan kebutuhan dalam pengembangan proyek perangkat lunak.

2. Pencarian Jurnal

Dalam pembuatan paper ini, kami mencari jurnal internasional dengan indeks ScienceDirect dan IEEE. adapun kata kunci yang kami gunakan untuk menemukan jurnal yang terkait adalah *change management*, *requirement change management* yang telah terpublikasi sejak tahun 2013.

### 3. Filterasi Jurnal

Dari hasil pencarian jurnal tersebut kami lakukan penyaringan dengan beberapa pertanyaan untuk menguji kualitas jurnal. Adapun pertanyaan tersebut antara lain:

Q1: Apakah jurnal tersebut terkait dengan change management?

Q2: Apakah jurnal tersebut terkait dengan requirement change management?

Q3: Apakah studi kasus yang digunakan dalam jurnal tersebut terkait dengan software development?

Q4: Apakah ada metode yang digunakan dalam jurnal tersebut?

Q5: Apakah ada alat bantu yang digunakan dalam jurnal tersebut?

### 4. Penilaian Jurnal

Berdasarkan dari hasil penyaringan jurnal, kami memberikan bobot nilai kepada masing-masing jurnal berdasarkan pertanyaan pada tahap filterisasi jurnal. Jurnal diberikan nilai satu apabila jurnal tersebut menjawab pertanyaan pada penyaringan jurnal, dan bobot kosong apabila tidak menjawab pertanyaan. Sehingga didapatkan total nilai yang dapat diberikan untuk satu jurnal maksimum sebesar lima. Hal ini dilakukan sebagai skala kuantitatif untuk mengurutkan jurnal berdasarkan korelasinya dengan pembahasan kami.

### 5. Klasifikasi Jurnal

Berdasarkan dari penyaringan jurnal tersebut, kami melakukan klasifikasi berdasarkan studi kasus, metode yang digunakan, dan alat bantu yang digunakan. Didapatkan 26 jurnal yang kemudian kami memilih 5 jurnal utama dan 21 jurnal pendukung sebagai acuan dari pembuatan *review paper* ini.

### 6. Kesimpulan

Berdasarkan klasifikasi tersebut kami melakukan pembahasan dan menarik sebuah kesimpulan dalam bentuk ringkasan mengenai penyebab manajemen perubahan, dampaknya, metode dan tools yang dapat digunakan.

## HASIL DAN PEMBAHASAN

Dengan menggunakan pendekatan review, didapatkan hasil yang sesuai dengan pendekatan review yang dilakukan. Dibawah ini merupakan hasil berdasarkan pendekatan review yang digunakan.

### RA 1 Menentukan Topik

*Review paper* ini menggunakan tipe dan tujuan *review* secara *summaries* (merangkum). Dalam melakukan *review paper*, digunakan beberapa tahap, tahapan yang pertama yaitu tahap menentukan topik. Dalam *review paper* ini topik yang diambil yaitu tentang manajemen perubahan dan agar lebih spesifik, topik yang

diambil tentang manajemen perubahan dalam pengembangan proyek perangkat lunak.

#### RA 2 Pencarian Jurnal

Pada tahap pencarian jurnal database yang digunakan adalah Sciencedirect dan IEEE. Jumlah jurnal yang dicari pada database ScienceDirect menggunakan kata kunci "Requirement Change Management" dapat dilihat pada Tabel 1 sedangkan dengan kata kunci "Change Management" dapat dilihat pada Tabel 2

**Tabel 1 Database Jurnal ScienceDirect "Requirement Change Management"**

Tahun	Jumlah Jurnal
2018	1
2017	6
2016	4
2015	8
2014	5
2013	2

Dari tabel diatas dapat disimpulkan bahwa jumlah jurnal dari tahun 2013 hingga tahun 2018 sebanyak 26 jurnal.

**Tabel 2 Database Jurnal ScienceDirect "Change Management"**

Tahun	Jumlah Paper
2018	542
2017	1007
2016	964
2015	1016
2014	915
2013	818

Dari tabel diatas dapat disimpulkan bahwa jumlah jurnal dari tahun 2013 hingga tahun 2018 sebanyak 5.262 jurnal. Sedangkan jumlah jurnal yang dicari pada database IEEE menggunakan kata kunci "Requirement Change Management" dapat dilihat pada Tabel 3 dan dengan kata kunci "Change Management" dapat dilihat pada Tabel 4.

**Tabel 3. Database Jurnal IEEE "Requirement Change Management"**

Tahun	Jumlah Paper
-------	--------------

2018	-
2017	1
2016	2
2015	-
2014	3
2013	5

Dari tabel diatas dapat disimpulkan bahwa jumlah jurnal dari tahun 2013 hingga tahun 2018 sebanyak 11 jurnal.

**Tabel 4. Database Jurnal IEEE “Change Management”**

Tahun	Jumlah Paper
2018	5
2017	49
2016	46
2015	38
2014	57
2013	46

Dari tabel diatas dapat disimpulkan bahwa jumlah jurnal dari tahun 2013 hingga tahun 2018 sebanyak 241 jurnal. Setelah tahap kedua selesai maka jurnal tersebut akan di filterasi sesuai dengan pertanyaan review.

#### RA 3 Filterasi Jurnal

Tahap ini berisi tentang hasil filterisasi untuk menguji kualitas jurnal yang telah dilakukan berdasarkan enam pertanyaan.

Q1: Apakah jurnal tersebut terkait dengan change management?

Berdasarkan populasi jurnal yang kami dapat dari database jurnal Sciencedirect dan IEEE, kami telah melakukan filterisasi jurnal yang membahas tentang *change management*. Kami melakukan filterisasi dengan melihat judul, kemudian abstrak dan kesimpulan. Apabila abstrak dan kesimpulan membahas tentang change management, maka selanjutnya isi dari jurnal tersebut akan dibahas.

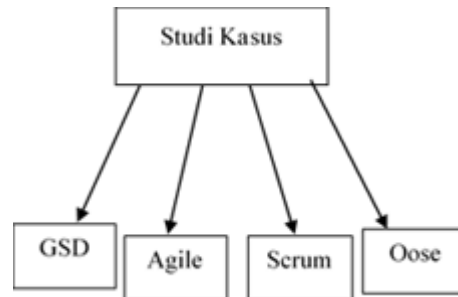
Q2: Apakah jurnal tersebut terkait dengan requirement change management?

Setelah jurnal dilakukan filterisasi sesuai dengan change management, maka lebih dispesifik kan lagi terkait dengan requirement change manajemen dalam bidang pengembangan proyek perangkat lunak. Jurnal yang sesuai dengan pendekatan ini menjadi semakin berkurang hingga didapat 26 jurnal.

Q3: Apakah studi kasus yang digunakan dalam jurnal tersebut terkait software development?



Terdapat beberapa studi kasus pada jurnal yang telah dipilih. Studi kasus dikategorikan berdasarkan kasus tersebut terjadi pada metode pengembangan perangkat lunak seperti apa. Kategori studi kasus dapat dilihat pada Gambar 2.

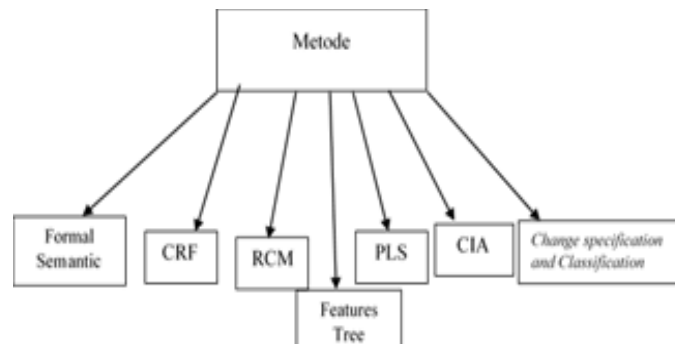


**Gambar 2 Studi Kasus**

Studi kasus yang didapatkan dari jurnal yang telah melewati tahap filterisasi sebanyak 4 studi kasus yang terjadi pada metode pengembangan perangkat lunak apa. Metode pengembangan yang didapatkan yaitu Global Software Development (GSD), Agile Software Development (ASD), Scrum, dan OOSE.

Q4: Apakah ada metode yang digunakan dalam jurnal tersebut?

Berdasarkan jurnal yang telah dipilih, didapatkan beberapa metode dalam mengatasi perubahan.

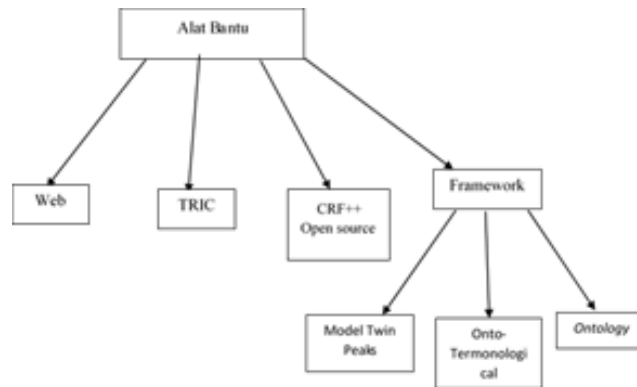


**Gambar 3 Metode Pada Jurnal**

Dari Gambar 3 terdapat 7 metode yang didapat dari jurnal yang telah dipilih. Dari 7 metode tersebut, 3 metode digunakan untuk analisis dampak akan adanya perubahan. 3 metode tersebut adalah metode formal semantic, metode *Conditional Random Field* (CRF), dan *Change Impact Analysis* (CIA). Sedangkan 4 metode lain yaitu *Requirement Change Management* (RCM), *Features Tree*, *Partial Least Square* (PLS), dan *Change Spesification and Classification* merupakan metode untuk membantu mengelola manajemen perubahan.

Q5: Apakah ada alat bantu yang digunakan dalam jurnal tersebut?

Terdapat beberapa alat bantu yang digunakan dalam jurnal yang telah dipilih.



**Gambar 4. Alat Bantu Pada Jurnal**

Pada Gambar 4. dapat dijelaskan bahwa pada 26 jurnal yang telah dipilih dapat diambil 4 alat bantu yaitu, web, *Tool Requirements Inferencing and Consistency Checking* (TRIC), *CRF++ open source*, dan framework. terdapat 3 framework yang digunakan sebagai alat bantu (1) Model Twin Peaks, Onto-terminological, dan ontology. 4 alat bantu diatas digunakan untuk memudahkan metode dalam mengatasi adanya perubahan.

#### RA 4 Penilaian Jurnal

Berdasarkan dari hasil pencarian jurnal yang terkait, kami memberikan bobot nilai kepada masing-masing jurnal untuk mempermudah kami dalam melihat keterkaitan jurnal dengan pokok bahasan kami. Adapun hasil dari pemberian bobot ini juga mempermudah kami dalam memilih jurnal utama dan jurnal pendukung.

**Tabel 5. Tabel Penilaian Jurnal**

No.	RA1	RA2	RA3	RA4	RA5	Bobot
1	1	1	1	1	0	4
2	1	1	1	1	1	5
3	0	1	1	1	0	3
4	0	1	1	1	1	4
5	1	0	0	1	1	3
6	0	1	1	1	1	4
7	1	1	1	1	1	5
8	1	1	1	0	1	4
9	1	1	1	1	0	4
10	1	1	1	1	1	5
11	1	1	1	1	1	5
12	1	0	0	1	1	3
13	1	1	1	1	1	5
14	1	0	0	1	1	3
15	1	1	1	1	1	5
16	1	0	0	1	0	2
17	1	0	0	1	0	2
18	1	0	0	1	0	2
19	1	1	0	1	0	3
20	1	0	1	1	0	3
21	1	1	1	1	0	4
22	1	0	0	1	1	3
23	1	0	1	1	1	4
24	1	0	0	1	0	2
25	1	0	1	1	1	4
26	1	1	1	1	0	4

**RA 5 Klasifikasi Jurnal**

Berdasarkan hasil dari proses pendekatan penelitian yang telah dilakukan, terdapat 26 jurnal yang terkait. dan dari 26 jurnal tersebut kami klasifikasikan kembali berdasarkan studi kasus, metode dan alat bantu yang digunakan, sehingga didapatkan 20 studi kasus dari 19 jurnal yang terkait dengan pengembangan perangkat lunak. Hasil klasifikasi jurnal dapat dilihat pada Tabel 6.

**Tabel 6. Tabel Klasifikasi Jurnal**

No	Studi Kasus	Metode	Alat Bantu
1	GSD	RCM	-tidak disebutkan-
2	ASD	Features Tree	Web Based
3	-tidak disebutkan-	Formal Semantic	-tidak disebutkan-
4	-tidak disebutkan-	CRF	CRF++
5	-tidak disebutkan-	PLS	Five Likert Chart
6	-tidak disebutkan-	RCM	-tidak disebutkan-
7	SCRUM	-tidak disebutkan-	Web Based
8	GSD	RCM	-tidak disebutkan-
9	GSD	RCM	-tidak disebutkan-
10	OOSE	RCM	UML
11	GSD	RCM	Ontology
12	GSD	RCM	-tidak disebutkan-
13	GSD	RCM	Spreadsheet
14	-tidak disebutkan-	Studi Kualitatif	-tidak disebutkan-
15	GSD	RCM	-tidak disebutkan-
16	GSD	RCM	-tidak disebutkan-
17	GSD	RCM	-tidak disebutkan-
18	-tidak disebutkan-	RCM	Framework : Twin Peaks Model
19	-tidak disebutkan-	Change Specification and Classification	Framework : Onto-Thermonological
20	ASD, SCRUM	Change Impact Analysis	-tidak disebutkan-

Berikut ini akan dilakukan pembahasan untuk menjawab empat pertanyaan penelitian tentang perubahan dalam pengembangan perangkat lunak. Masing - masing pertanyaan akan dibahas dibawah ini.

RQ1: Apa saja hal – hal yang dapat menyebabkan manajemen perubahan?

Hal yang dapat menyebabkan manajemen perubahan kebutuhan adalah terjadinya perubahan itu sendiri. Kebutuhan user dalam pengembangan perangkat lunak menjadi lebih rumit dan dapat berubah setiap saat untuk mencapai target operasi bisnis sesuai dengan visi dan misi dari perusahaan. banyak faktor alasan terjadinya perubahan kebutuhan: permintaan dari pengguna, bertambahnya pemahaman terhadap kebutuhan klien, tersedianya teknologi baru, dan khususnya di perkembangan dunia saat ini, perubahan struktur organisasi perusahaan sebagai respon terhadap kompetisi bisnis. Perubahan kebutuhan adalah aktivitas pengembangan perangkat lunak yang tidak dapat dihindari dan dapat terjadi karena perubahan dalam kebutuhan dari pengguna, peningkatan pemahaman tentang kebutuhan pemangku kepentingan, restrukturisasi organisasi pelanggan, dan ketersediaan teknologi baru (Ali dan Lai, 2016).

Perubahan dalam suatu proyek dapat terjadi baik sebelum, saat maupun setelah proses pengembangan perangkat lunak. Secara umum, hal yang dapat menyebabkan manajemen perubahan, adalah terjadinya suatu perubahan yang kritis dan dapat berdampak kepada keberhasilan proyek. Perubahan kebutuhan merupakan salah satu faktor yang sulit untuk dihindari, dan adapun penyebab terjadinya perubahan kebutuhan diantaranya adalah : (1) tahap elistitasi kebutuhan yang kurang dalam dan tepat sasaran dalam mencari kebutuhan, (2) adanya perubahan kebutuhan dikarenakan bertambahnya pemahaman dari kebutuhan user oleh tim pengembang maupun user itu sendiri , (3) tersedianya teknologi yang baru, (4) terjadinya restrukturisasi organisasi baik dari client maupun tim pengembang, (5) munculnya regulasi baru baik dari perusahaan maupun pemerintah.

RQ 2: Bagaimana Dampak Dari Perubahan?

Dampak dari perubahan kebutuhan, elemen desain dan kode sumber perlu dianalisis untuk menentukan bagian dari sistem perangkat lunak yang akan diubah. Dengan dukungan adanya dari perubahan alternatif yang diidentifikasi dan eliminasi dari dampak positif yang salah serta merubah pengecekan secara konsisten (Goknill et al., 2014). Dengan adanya pengecekan yang secara konsisten alternatif dari perubahan kebutuhan dapat di prediksi dan dapat mengurangi dampak yang merugikan yang biasanya sering terjadi dalam industri.

Perubahan kebutuhan dari tuntutan pengguna, pemangku kepentingan, dan organisasi dapat mengakibatkan total biaya, kualitas, dan jadwal akan sangat dipengaruhi skarena adanya kebutuhan perubahan. adanya perubahan kebutuhan dinilai sebagai salah satu penyebab utama dalam kegagalan proyek perangkat lunak (Ali dan Lai, 2016). Biaya akan melebihi dari biaya awal, kualitas dari perangkat lunak

yang dihasilkan kemungkinan akan turun kemudian seringnya jadwal akan melebihi jadwal awal yang telah ditetapkan sebelumnya.

Kebutuhan perubahan adalah kegiatan yang tidak dapat dihindari dan dapat dilakukan terjadi karena perubahan tuntutan pengguna, peningkatan pemahaman dari para pemangku kepentingan, organisasi pelanggan, visi proyek, kebutuhan spesifikasi dan ketersediaan teknologi (Ali dan Lai, 2014). Kebutuhan akan selalu berubah dalam suatu pengembangan perangkat lunak disebabkan oleh pengguna, kebijakan, dan perubahan teknologi (Akhtar, Motla, Aslam, dan Jamal, 2014). Untuk menganalisis dampak dari adanya perubahan, beberapa *paper* menggunakan metode dalam analisis dampak. Dibawah ini merupakan beberapa metode tersebut:

a. Formal Semantics

Metode *Formal semantics* dengan tiga aktifitas untuk analisis dampak. Yang pertama, ahli dari kebutuhan perubahan mengusulkan perubahan dengan klasifikasi yang ada sebelumnya, sebelum dilakukan perubahan yang sebenarnya. Kemudian yang kedua ahli harus mengidentifikasi apa saja yang akan mendapat dampak dari perubahan. dan yang ketiga yaitu, mengidentifikasi perubahan – perubahan apa saja yang mungkin terjadi.

Dalam penelitian Goknill et al. (2014) analisis dampak perubahan menggunakan klasifikasi perubahan kebutuhan berdasarkan asumsi tentang struktur kebutuhan tekstual yang sangat umum. Jenis perubahan dianalisis berdasarkan efeknya menggunakan rumus untuk mewakili kebutuhan.

b. *Conditional Random Field* (CRF)

CRF adalah sebuah metode statistik yang dapat digunakan untuk mengukur secara kuantitatif interaksi antara pengguna dan sistem yang seringkali mengarah kepada potensi kebutuhan pengguna dan persyaratan. Dengan menganalisa pola perilaku *run-time* pengguna, pakar dibidangnya dapat membuat prediksi cepat tentang bagaimana niat pengguna berubah, dan secara tepat waktu mengajukan peningkatan sistem, atau perbaikan untuk membantu mengatasi kebutuhan yang muncul. Dalam penelitian Xie et al. (2017), metode CRF digunakan untuk mendukung *Software Evolution*.

c. *Change Impact Analysis* (CIA)

Change Impact Analysis (CIA) menentukan efek potensial pada sistem yang dihasilkan dari perubahan yang diusulkan. CIA dapat digunakan untuk memprediksi efek dari perubahan sebelum diimplementasikan, memberikan perkiraan upaya / biaya untuk mengimplementasikan perubahan, serta potensi risiko yang terlibat dalam membuat perubahan. Analisis ini dapat digunakan untuk membuat keputusan evolusi yang lebih baik seperti apakah perubahan harus dilakukan berdasarkan kelayakan ekonomi evolusi perangkat lunak atau risiko lainnya seperti degradasi sistem perangkat lunak. Prosesnya, implementasi dari teknik CIA ini, terdiri dari dua langkah utama yang dijelaskan di bawah ini: (1) Memberikan perubahan dalam fitur (menambahkan menghapus atau updating), algoritma berbasis traceability menentukan (i) keputusan susunan desain pertama yang terlibat dengan fitur yang

akan diubah, (ii) DD susunan ke-n yang bergantung pada DD susunan pertama, dan (iii) elemen arsitektur susunan pertama (PPC, komponen, dan konektor) yang terlibat dalam masing-masing (urutan pertama dan n) DD. Algoritma ini melintasi tautan penelusuran yang menjembatani fitur dan elemen arsitektur, dan hubungan ketergantungan antara keputusan desain. (2) memberikan perubahan dalam suatu arsitektur kerja yang menyadari perubahan dalam fitur (Jessica Díaz et al. 2013)

RQ 3: Apa metode yang digunakan untuk mengatasi perubahan?

Dalam mengatasi perubahan yang sering terjadi dibutuhkan manajemen perubahan yang tepat. dalam mengelola kebutuhan perubahan. Selain manajemen yang tepat, metode yang digunakan juga harus mendukung manajemen perubahan kebutuhan. Berikut akan dijelaskan beberapa metode yang dapat digunakan dalam menangani perubahan.

a. *Requirement Change Management (RCM)*

Metode RCM sering digunakan dalam kasus yang termasuk dalam kasus skala besar. Proses *Requirement Change Management* terdiri dari tiga tahapan utama, yakni (1) memahami, (2) analisa, dan (3) menyelesaikan perubahan kebutuhan. Dalam *paper review* ini beberapa penelitian menggunakan metode RCM.

Menurut Ali dan Lai (2014) bahwa RCM merupakan salah satu proses dari kegiatan yang paling serius dan menimbulkan kesulitan dengan tim pengembangan perangkat lunak. Kurangnya penerapan RCM yang tepat dapat menyebabkan perangkat lunak mengalami kegagalan dan kehilangan bisnis. Proses RCM menjadi sangat sulit karena tantangan komunikasi. komunikasi merupakan salah satu masalah utama dalam proyek perangkat lunak yang global secara jarak geografis dan sosial-budaya.

b. *Features Tree*

Metode ini dilakukan dalam dua tahapan utama, (1) membangun bentuk awal dari *Features Tree* dan (2) manajemen perubahan kebutuhan. Bentuk awal dari *Features Tree* berisikan gambaran kebutuhan yang mungkin didapatkan dari pengembangan perangkat lunak. Kemudian tim pengembang melakukan manajemen perubahan kebutuhan dengan cara melakukan persetujuan terhadap perubahan yang baru atau tetap menggunakan kebutuhan yang sudah teridentifikasi sebelumnya. Adapun metode ini dapat dilakukan dengan tiga aktifitas: (1) modifikasi terhadap perubahan dengan cara menambah, menghapus, atau memperbaiki, (2) mengontrol perubahan dengan cari menyetujui perubahan, atau tetap dengan kebutuhan yang sudah ada sebelumnya, dan (3) mencari hubungan ketergantungan antar kebutuhan

dalam hal implikasi dan pengecualian (Lloyd et al., 2017). Metode ini dapat digunakan dengan alat bantu web.

c. *Partial Least Square (PLS)*

Metode dengan pendekatan berbasis variansi. Metode ini digunakan dalam penelitian Ruhaya Ab. Aziz dan Bernard Wong pada tahun 2015 yang meneliti dampak RRK (Requirement Relationship Knowledge) terhadap perubahan kebutuhan dan keberhasilan proyek. Dalam penelitian ini, Partial Least Square (PLS) digunakan karena alasan berikut: (1) penelitian tentang hubungan kebutuhan yang relatif baru, dan (2) tidak ada model pengukuran yang sudah tersedia. PLS bisa menjadi teknik yang cocok untuk digunakan ketika fenomena yang akan diteliti masih relatif baru (Aziz dan Wang, 2015).

d. *Change specification and Classification*

Metode ini merupakan suatu pendekatan yang dapat dilakukan untuk mendukung komunikasi dan elisitasi perubahan. Kerangka kerja Specification berdasarkan konsep onto-terminological yang merupakan penggabungan dari *ontology* dan *terminology*. Hal ini dibangun dengan menggunakan abstraksi dari Matriks Pertanyaan Sasaran yang mewakili fungsi linguistik (*terminology*) sementara deskripsi abstraksi dari Kerangka Kerja Sumber Daya (mewakili *ontology*) digunakan untuk menguraikan koneksi logis dari ketentuan.

Pada perubahan permintaan, pendekatan ini mampu: (1) membantu komunikasi perubahan dari *requester* (bagian bisnis) kepada implementer (bagian IT) dalam satu tingkatan operasional, (2) mengklarifikasikan alasan dari perubahan, yang menghasilkan identifikasi terhadap tujuan dari perubahan, dan (3) mampu mengidentifikasi tipe dari perubahan melalui klasifikasi perubahan dan menyediakan pedoman dasar untuk menggabungkan perubahan kedalam desain sistem. Dengan pendekatan ini, kita dapat menjabarkan perubahan kebutuhan berdasarkan pedoman yang jelas dan juga menyediakan bahasa penengah antara bagian IT dan bagian bisnis (Jayatilleke dan Lai, 2013).

RQ 4: Apa alat bantu yang digunakan untuk mengatasi perubahan?

Untuk mendukung dan membantu manajemen dalam mengelola perubahan diperlukan alat bantu untuk memfasilitasi dan mengelola model kebutuhan dan perubahan dalam semua fase. Ada beberapa alat bantu yang dapat digunakan dalam mendukung metode - metode perubahan.

a. Web

Web merupakan alat bantu dengan pertimbangan yang cepat dalam distribusi. Dengan lokasi geografis yang berbeda antara tim pengembang, komunikasi dapat tetap dapat dilakukan dan dapat dikelola. Web dimaksudkan untuk memfasilitasi dan mengelola model kebutuhan dan perubahan dalam semua fase mulai dari inisiasi proyek, melacak kebutuhan dan perubahannya. Dengan alat ini dapat memungkinkan

manajer proyek untuk melacak status dalam semua fase pengembangan (Llyod et al., 2017). Untuk membantu penelitian Alsemi dan Yeoh 2015 tentang bagaimana mengelola perubahan product backlog pada proyek scrum digunakan web sebagai alat bantu untuk kuisisioner dengan partisipasi dari proyek scrum diseluruh dunia.

b. *Tool Requirements Inferencing and Consistency Checking (TRIC)*

TRIC mempunyai fitur untuk analisis dampak perubahan dalam kebutuhan. Fitur - fitur dari TRIC adalah sebagai berikut: (1) Mengusulkan perubahan (2) Menyebarkan perubahan (3) Menampilkan perubahan yang tidak konsisten (4) Implementasi usulan perubahan dalam model kebutuhan (5) Memprediksi dampak dari usulan perubahan. Dalam hal mengusulkan perubahan TRIC menyediakan GUI untuk mengusulkan perubahan yang berisikan daftar semua kebutuhan. Setelah perubahan diusulkan alat ini akan mencantumkan nama ahli yang dapat menyebarkan perubahan. Kemudian alat akan menentukan dampak perubahan dengan cara mengklik maka alat akan memunculkan dampak alternatif dari perubahan (Goknill et al., 2014)

c. *CRF++ open source*

CRF++ merupakan alat bantu yang bersifat open-source, yang digunakan untuk membuat model CRF. dalam CRF++ terdapat fitur untuk mendesain, dan secara otomatis dapat diproses berdasarkan fitur templatnya. Pada penelitian yang dilakukan oleh Haihua Xie et al. (2017), CRF++ digunakan sebagai alat bantu untuk membuat model CRF guna membuat pendekatan untuk memprediksi perubahan kebutuhan pengguna dalam rangka untuk *software service evolution*.

d. *Framework*

Terdapat 3 framework yang dapat digunakan sebagai alat bantu, yaitu

(1) Framework: Model Twin Peaks

Framework Model twin peaks digunakan untuk menganalisis perubahan kebutuhan dan dampak dari perubahan kebutuhan dari perancangan arsitektur perangkat lunak. Framework ini sangat membantu dalam mengelola arsitektur perangkat lunak (Akhtar et al., 2014).

(2) Framework: Onto-Termonological

Framework Onto-Termonological merupakan kombinasi antara Ontology dan Termonological. Sehingga dalam framework ini menguraikan baik linguistik dan hubungan logis dari metode ini.

(3) Framework: *Ontology*



Dalam hal ini, *ontology* dapat digunakan sebagai alat bantu yang memberikan solusi untuk meminimalis ambiguitas antar pemangku kepentingan ketika kebutuhan berubah secara dinamis. Pada penelitian yang dilakukan oleh Asma Kathoon, et al. (2013), *ontology* digunakan dalam tahapan RCM dengan menggunakan *Unified Modelling Language (UML)* untuk membuat konsep dari perubahan kebutuhan dalam *Global Software Development (GSD)* dan melakukan proses pengembangan *ontology* dengan menggunakan *ontology tool protégé*. *Ontology* memiliki tiga komponen utama, yaitu instances, properties dan concept. Instances menunjukkan data proyek tertentu, properti menampilkan hubungan antara konsep dan konsep mewakili kelas atau konsep dari domain sebuah proyek.

## KESIMPULAN

Dari banyaknya jurnal yang kami dapatkan, diketahui bahwa dalam 19 jurnal yang terkait, terdapat 20 studi kasus. Dimana dalam sembilan belas studi kasus tersebut, terdapat empat tipe pengembangan perangkat lunak yaitu Global Software Development, Agile, Scrum dan OOSE. Sembilan jurnal membahas mengenai pengembangan perangkat lunak GSD, dan seluruhnya menggunakan metode Requirement Change Management (RCM). Terdapat satu jurnal membahas mengenai pengembangan perangkat lunak secara Agile, dan metode yang digunakan adalah features tree. Satu jurnal juga menyebutkan pengembangan perangkat lunak secara OOSE dengan menggunakan metode RCM. Dan terdapat satu jurnal membahas mengenai pengembangan software secara Scrum yang menekankan penggunaan web-based sebagai alat bantu dalam manajemen perubahan.

Beberapa metode dan alat bantu yang telah didefinisikan pada bab hasil, dapat diambil kesimpulan berdasarkan studi kasus, metode, dan alat bantu. Dari banyaknya jurnal yang menggunakan studi kasus pada pengembangan perangkat lunak dengan menggunakan Global Software Development (GSD) dan metode yang digunakan adalah Requirement Change Management (RCM). Metode RCM digunakan pada 11 jurnal. Dari banyaknya jurnal yang menggunakan metode RCM maka dapat disimpulkan bahwa metode Requirement Change Management (RCM) merupakan metode yang paling baik dalam menangani dan mengelola manajemen perubahan. Tidak banyak jurnal yang menggunakan alat bantu dalam mendukung metode, tetapi terdapat alat bantu web yang digunakan pada 2 studi kasus. Web digunakan karena dianggap lebih mudah untuk mengelola data kualitatif dan kuantitatif.

## DAFTAR PUSTAKA

- Ali, N., & Lai, R. (2016). A method of requirements change management for global software development. *Information And Software Technology*, 70, 49-67. <https://doi.org/10.1016/j.infsof.2015.09.005>
- Lloy, D., Moawad, R., & Kadry, M. (2017). A supporting tool for requirements change management in distributed agile development. *Future Computing And Informatics Journal*, 2(1), 1-9. <https://doi.org/10.1016/j.fcij.2017.04.001>
- Goknil, A., Kurtev, I., van den Berg, K., & Spijkerman, W. (2014). Change impact analysis for requirements: A metamodeling approach. *Information And Software Technology*, 56(8), 950-972. <https://doi.org/10.1016/j.infsof.2014.03.002>

- Xie, H., Yang, J., Chang, C., & Liu, L. (2017). A statistical analysis approach to predict user's changing requirements for software service evolution. *Journal Of Systems And Software*, 132, 147-164. <https://doi.org/10.1016/j.jss.2017.06.071>
- Martinsuo, M., & Hoverfält, P. (2018). Change program management: Toward a capability for managing value-oriented, integrated multi-project change in its context. *International Journal Of Project Management*, 36(1), 134-146. <https://doi.org/10.1016/j.ijproman.2017.04.018>
- Aziz, R., & Wong, B. (2015). The Interplay between Requirements Relationships Knowledge and Requirements Change towards Software Project Success: An Assessment Using Partial Least Square (PLS). *Procedia Computer Science*, 46, 732-741. <https://doi.org/10.1016/j.procs.2015.02.138>
- Khan, A., Basri, S., Dominic, P., & Fazal-e-Amin. (2012). A process model for Requirements Change Management in collocated software development. *2012 IEEE Symposium On E-Learning, E-Management And E-Services* (pp. 1-6). IEEE. <https://doi.org/10.1109/is3e.2012.6414949>
- Alsalemi, A., & Yeoh, E. (2015). A survey on product backlog change management and requirement traceability in agile (Scrum). *2015 9Th Malaysian Software Engineering Conference (Mysec)*. IEEE. <https://doi.org/10.1109/mysec.2015.7475219>
- Waqar Hussain, Hussain, W. (2016). Reflections on Requirements Change Management in Global Software Development: A Multiple Case Study. *2016 IEEE 11Th International Conference On Global Software Engineering Workshops (ICGSEW)* (pp. 77-79). IEEE. <https://doi.org/10.1109/icgsew.2016.25>
- Tomyim, J., & Pohthong, A. (2016). Requirements change management based on object-oriented software engineering with unified modeling language. *2016 7Th IEEE International Conference On Software Engineering And Service Science (ICSESS)* (pp. 7-10). IEEE. <https://doi.org/10.1109/icsess.2016.7883005>
- Khatoon, A., Motla, Y., Azeem, M., Naz, H., & Nazir, S. (2013). Requirement change management for global software development using ontology. *2013 IEEE 9Th International Conference On Emerging Technologies (ICET)* (pp. 1-6). IEEE. <https://doi.org/10.1109/icet.2013.6743519>
- Juszczuk, M., Tomana, A., & Bartoszek, M. (2016). Current Issues of BIM-based Design Change Management, Analysis and Visualization. *Procedia Engineering*, 164, 518-525. <https://doi.org/10.1016/j.proeng.2016.11.653>
- Khan, A. A., Keung, J., Hussain, S., Kwabena, Bennin, E., (2015). Effects of Geographical, Socio-cultural and Temporal Distances on Communication in Global Software Development during Requirements Change Management. *2015 International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE)* (pp. 159-168). IEEE.
- Schuh, G., Gartzten, T., Soucy-Bouchard, S., & Basse, F. (2017). Enabling Agility in Product Development through an Adaptive Engineering Change Management. *Procedia CIRP*, 63, 342-347. <https://doi.org/10.1016/j.procir.2017.03.106>
- Hussein, W., & Clear, T. (2014). Spreadsheets as Collaborative Technologies in Global Requirements Change Management. *2014 IEEE 9th International*

- Conference on Global Software Engineering* (pp. 74-83). IEEE. <https://doi.org/10.1109/ICGSE.2014.25>.
- Schuh, G., Riesener, M., Tönnies, C., & Aleksic, S. (2017). Technical change management for the maintenance of product platforms. *Procedia CIRP*, 60, 458-463. <https://doi.org/10.1016/j.procir.2017.01.025>
- Gerbec, M. (2017). Safety change management—A new method for integrated management of organizational and technical changes. *Safety Science*, 100, 225-234. <https://doi.org/10.1016/j.ssci.2016.07.006>
- Al-Moosa, N. H., & Sharts-Hopko, N. (2017). Using change management to redesign Oman' s health professions education sector. *Health Professions Education*, 3(2), 108-112. <https://doi.org/10.1016/j.hpe.2016.09.001>
- Hussain, W., Zowghi, D., Clear, T., MacDonell, S., & Blincoe, K. (2016, September). Managing requirements change the informal way: When saying 'no' is not an option. *2016 IEEE 24th International Requirements Engineering Conference (RE)* (pp. 126-135). IEEE. <https://doi.org/10.1109/RE.2016.64>
- Khan, A. A., Basri, S., & Dominc, P. D. D. (2014). A proposed framework for communication risks during RCM in GSD. *Procedia-Social and Behavioral Sciences*, 129, 496-503. <https://doi.org/10.1016/j.sbspro.2014.03.706>
- Ali, N., & Lai, R. (2014, November). Managing requirements change in global software development. *2014 International Conference on Data and Software Engineering (ICODSE)* (pp. 1-5). IEEE. <https://doi.org/10.1109/ICODSE.2014.7062700>
- Serapelo, T. E., Erasmus, L., & Pretorius, J. H. (2017, July). Engineering Change Management Impact on Project Success within a South African Petrochemical Company. In *2017 Portland International Conference on Management of Engineering and Technology (PICMET)* (pp. 1-8). IEEE. <https://doi.org/10.23919/PICMET.2017.8125469>
- Akhtar, A., Motla, Y. H., Aslam, H., & Jamal, M. (2014, June). Role of requirement change in software architecture using Twin Peaks Model. In *2014 IEEE 5th International Conference on Software Engineering and Service Science* (pp. 174-177). IEEE. <https://doi.org/10.1109/ICSESS.2014.6933539>
- Lines, B. C., Sullivan, K. T., Smithwick, J. B., & Mischung, J. (2015). Overcoming resistance to change in engineering and construction: Change management factors for owner organizations. *International Journal of Project Management*, 33(5), 1170-1179. <https://doi.org/10.1016/j.ijproman.2015.01.008>
- Jayatilleke, S., & Lai, R. (2013, June). A method of specifying and classifying requirements change. In *2013 22nd Australian Software Engineering Conference* (pp. 175-180). IEEE. <https://doi.org/10.1109/ASWEC.2013.29>
- Díaz, J., Pérez, J., Garbajosa, J., & Yagüe, A. (2013, January). Change-impact driven agile architecting. In *2013 46th Hawaii international conference on system sciences* (pp. 4780-4789). IEEE. <https://doi.org/10.1109/HICSS.2013.127>