

TOWARDS IDENTIFICATION AND REFINEMENT BY USING PATTERN-BASED APPROACH

¹Rizwana Yasmeen, ²Dr. Mudasir Mahmood

^{1&2}Department of Computer Science, Gomal D.I.K, Pakistan

¹rizwanakhan532@yahoo.com, ²mudasir@gu.edu.pk

Keywords

Identification, requirements, refinement, patterns, MSDA (modern software development approaches)

Article History

Received on 29 March, 2026

Accepted on 14 April, 2026

Published on 16 April, 2026

Copyright @Author

Corresponding Author:

Abstract

In contemporary (Modern computing systems support the execution of complex business processes through advanced software engineering methods. One of the key challenges in this context is ensuring that requirements are properly identified and refined to reflect the trust expectations of end-users throughout the software development life cycle. Trust has become an essential quality attribute for contemporary applications, making it necessary to integrate user concerns into early design stages. This research introduces a structured set of user-friendly questions aimed at helping end-users clearly express their trust-related needs. In addition, a pattern-based strategy is applied to the requirements engineering phase, offering practitioners a systematic approach to capture, evaluate, and address trust concerns as part of the development process.



INTRODUCTION

The process of identifying and refining requirements has become a critical challenge in modern software development approaches. At the same time, business service providers increasingly emphasize value creation, aiming not only to enhance the overall experience and satisfaction of their customers but also to maximize the advantages gained by various stakeholders connected to the organization. [Lepak and Taylor 2007].

Trustworthiness refers to a set of fundamental system qualities that directly shape the degree of confidence users place in a system. These properties must align with specific requirements to ensure reliability and credibility. While in some context's trustworthiness has been equated with concepts like security or dependability, it is in fact broader, encompassing but not limited to these dimensions. The concept remains somewhat ambiguous in academic discussions, as scholars often differ in how they define and apply it, leading to variations in interpretation across disciplines.) [Nazila and Maritta 2016].

The use of structured patterns to design clear and user-friendly questions enables end-users to articulate their trust concerns in a simple and meaningful way. This approach helps present

requirements in an easily interpretable form, minimizes ambiguity, encourages reusability, and guides requirements engineers in capturing all essential information without omissions. In many organizations, a process-oriented method is adopted to identify activities that contribute to value creation. Such methods are specifically developed to highlight opportunities for improving quality while simultaneously reducing time and effort in their application. [Grosskopf et al. 2009].

In our proposed pattern-based approach, end-users actively share their requirements and provide suggestions regarding existing problems and potential solutions. A pattern represents a problem-solution pair that can be reused in similar situations, accompanied by guidance on its application in new contexts. It is important to highlight that the formal definition of a pattern varies across different studies and literature.

There are three main types of patterns:

1. An architectural pattern, which spans across multiple software subsystems.
2. A design pattern, which operates within a subsystem but remains independent of the programming language.
3. An idiom, which is a low-level pattern specific to a particular programming lang

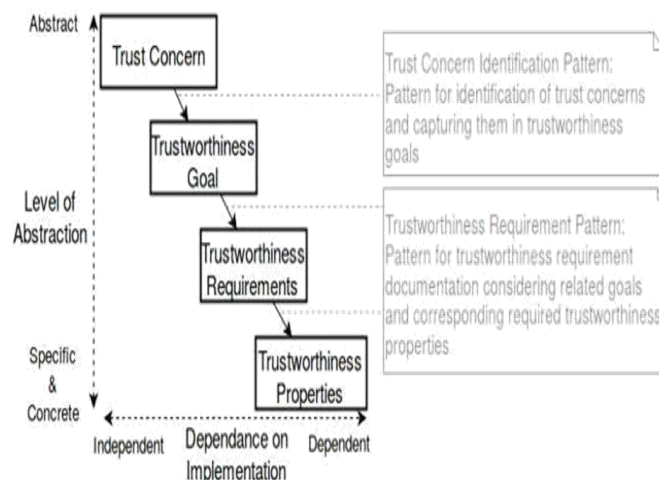


Fig.1: Relation between different concepts from trust to trustworthiness

LITERATURE REVIEW

A comprehensive review of existing literature reveals that researchers have employed multiple approaches to identify and refine trustworthiness requirements. The following section presents a

concise overview of the most relevant studies and highlights the significant contributions made by different scholars in this area.

Da Silva et al. (2015) highlighted the application of use cases as an effective technique for

requirements modeling. In their work, they introduced structured use case patterns that consist of several essential components, including the pattern's name, the context in which it is applied, the problem being addressed, the proposed solution, practical examples, possible consequences, connections with other patterns, and documented instances of use. This structured approach helps standardize requirement modeling and facilitates clearer communication among stakeholders.

Pohl (2010) conducted a study focusing on the use of patterns for representing scenarios in a well-structured textual format. His work emphasized the role of requirement templates, which not only capture standard scenarios but also account for potential exception cases, thereby ensuring a more comprehensive representation of system requirements.

In their study, del Río-Ortega et al. (2012) introduced a set of patterns designed to support the definition of process performance indicators. To enhance this framework, they developed an ontology-based linguistic pattern grounded in formal descriptive logic. This approach enables not only the structured representation of performance indicators but also facilitates automated reasoning and knowledge inference. As a result, it becomes possible to identify and analyze the interconnections among various key performance indicators within business process management.

In their study, Alebrahim et al. (2014) present a systematic method for assessing software quality through domain knowledge modeling. Their approach emphasizes the use of patterns to represent domain knowledge effectively, while also examining system requirements to minimize the risk of producing inaccurate specifications.

Romi S. Wahono and Behrouz H. Far, in their work "A Framework for Object Identification and Refinement Process in Object-Oriented Analysis and Design", propose a systematic method for deriving and refining objects from software requirements. Their approach leverages Object-Based Formal Specification (OBFS) to convert requirement statements into precise and structured objects. By applying this framework, the study ensures

greater clarity, accuracy, and consistency in the initial phases of software design, thereby improving the overall quality of object-oriented analysis and design.

In their 2013 study titled "Refining Scheduling Policies with Genetic Algorithms," Emin Ogur and Mehmet E. Aydin proposed a novel method for improving scheduling efficiency. The researchers focused on refining policy structures by optimizing the sequence of dispatching rules (DRs) applied within a specific time frame. Using a Genetic Algorithm (GA), they iteratively adjusted and evolved these sequences to identify configurations that yielded better scheduling outcomes. Their findings highlight the effectiveness of GA in systematically enhancing scheduling policies and achieving improved performance.

Maintaining the Integrity of the Specifications

Preserving the original formatting of this template is essential to uphold the integrity of the specifications. The layout, including margins, fonts, spacing, and column dimensions, has been purposefully designed and should remain unaltered. While some settings—such as the slightly wider top margin—may seem unconventional, they are deliberately applied to achieve consistency across the complete collection of papers. These adjustments are not intended for individual presentation but to ensure uniform appearance within the compiled proceedings. Therefore, it is crucial to retain all formatting features exactly as provided.

RESEARCH METHODOLOGY

To achieve the research objectives and address the proposed questions, this study adopted a **quantitative survey method**. The target population primarily consisted of employees working in large organizations where modern IT-based approaches, including pattern recognition techniques and advanced software development practices, are commonly applied. A **convenience sampling technique** was employed due to constraints of time and resources, meaning that participants were selected based on accessibility rather than random sampling. This approach allowed the researchers to collect relevant data within the available limitations. The sampling

frame included organizations managing sensitive information and large databases, with a considerable number of technical staff engaged in daily operations.

Protocol for search process

The search process begins with the development of a well-defined protocol based on the research questions. This protocol serves as a roadmap, detailing each step of the review. It includes the formulation of the search strategy, the criteria for including or excluding studies, the approach for identifying and selecting relevant literature, and the procedures for extracting, recording, and systematically organizing the necessary information.

SOURCES FOR LITERATURE SEARCH

The following databases were explored to identify relevant primary studies.

ACM <http://dl.acm.org/>

IEEE

<http://ieeexplore.ieee.org/Xplore/home.jsp>

Springer <http://link.springer.com/>

Inclusion criteria

The sources for the literature search were selected based on clearly defined inclusion criteria. Research papers were considered relevant if they contained specific keywords such as “Identification,” “Refinement,” “Requirements,” and “MSDA.” In addition, studies with titles closely related to the theme of “Identification and Refinement of Requirements” were prioritized. To ensure the review remained current and focused, only publications released between 2012 and 2018 were included.

Exclusion criteria

I excluded the papers based on the following points:

- Research work that does not fall within the defined time period.
- Studies in which the primary focus is not related to identification and refinement.

- Information obtained from unreliable resources (such as Wikipedia, encyclopedias, or digital magazine articles).

Research objective for the study

This research is designed with two primary objectives.

RO 1: To develop a set of comprehensive and user-friendly questions that enable end-users to clearly express and communicate their trust-related concerns.

RO 2: To identify, refine, and evaluate the trust concerns raised by end-users, while examining how these concerns influence and shape the overall requirements.

OVERVIEW OF MY METHOD

This section provides a brief overview of the three proposed patterns introduced in this study. These patterns are designed to support the process of analyzing and addressing trust-related aspects in system development. The patterns include:

1) **Documentation Pattern** - which outlines a structured approach for recording essential details.

2) **Trust Concern Identification Pattern** - which helps in recognizing and categorizing potential trust issues.

3) **Trustworthiness Requirement Pattern** - which defines the necessary conditions to ensure the reliability and credibility of the system.

This pattern-based approach helps to systematically refine and document the functional as well as non-functional requirements that satisfy the identified trustworthiness goals. We develop patterns to aid the documentation of trustworthiness requirements. The provided patterns are used during the requirement engineering phase to support the requirements engineer in documenting trustworthiness requirements by using linguistic templates.



Fig.2: Relation between the developed patterns and the process of applying them

Documentation patter

It A documentation pattern generally serves as a structured approach for presenting trust and trustworthiness. Its essential elements usually include the integration of goal models and process models, which together provide clarity, consistency, and reliability in the representation of trust-related information.

Context:

This This approach plays a crucial role in the requirements engineering phase, where the identification of trustworthiness requirements begins with goal-oriented models and is later elaborated through detailed business process models.

Problem and forces:

Without If trustworthiness requirements are not defined clearly and in an easily interpretable manner, there is a high risk that developers may misinterpret them and implement them incorrectly. To address this challenge, we introduce a structured pattern that allows trustworthiness requirements to be documented systematically and refined iteratively.

Solution:

Trust issues are recognized and managed in goal models through the inclusion of trustworthiness objectives. These objectives are progressively refined within the models, alongside business process models, to ensure the intended goals are achieved. By integrating goal models with business process models, it becomes clear how specific processes contribute to fulfilling the trustworthiness objectives.

Consequences:

The implementation of the suggested patterns leads to well-organized, transparent, and logically connected trustworthiness specifications, guaranteeing accuracy, thoroughness, and uniformity.

Trust concerns identification pattern

This pattern focuses on recognizing the trust-related issues of end-users and translating them into specific trustworthiness objectives that can be integrated within the software development process.

Context:

The trust concerns identification pattern plays an important role in addressing the higher-order documentation pattern by systematically recognizing potential trust issues. This pattern also assists in formulating and recording trustworthiness objectives that are necessary to respond to the identified concerns, ensuring that the system or process remains reliable and aligned with security and confidence requirements.

Problem and forces:

The main challenge lies in recognizing and categorizing trust concerns through a clear, systematic, and structured approach.

TABLE 1

This requires identifying underlying issues, mapping behavioral patterns, and analyzing external as well as internal forces that influence trust.

Solution:

Table 1 presents the suggested framework designed to assist requirement engineers in recognizing the trust-related concerns of end-users. The upper section of this framework is structured as a questionnaire, which needs to be completed by the end-user. The lower section, on the other hand, is reserved for inputs to be provided by the requirement engineer.

Consequences:

The end-user provides responses to the given questions. According to these responses, the requirement engineer is able to determine and choose the most appropriate trustworthiness goal.

Name	What is the name or identifier of the end-user?
Gender	Female <input type="checkbox"/> Male <input type="checkbox"/>
The reason to use the product	What are the reasons to use this product?
Motivation	What objectives does the end-user follow?
Please mark the following statements with yes or no:	
Yes <input type="checkbox"/> No <input type="checkbox"/>	I usually trust a person until there is a reason not to.
Yes <input type="checkbox"/> No <input type="checkbox"/>	I am afraid to lose control over my personal information.
Yes <input type="checkbox"/> No <input type="checkbox"/>	I am concerned about how my personal information is collected and used by digital applications and services.
Yes <input type="checkbox"/> No <input type="checkbox"/>	Organisations handle the personal information they collect about me in a proper way.
Yes <input type="checkbox"/> No <input type="checkbox"/>	Existing laws and organisational practices adequately protect me from online risks.
Yes <input type="checkbox"/> No <input type="checkbox"/>	I am concerned that legal measures such as the ability to delete your personal information are not put into place for increasing control.
Yes <input type="checkbox"/> No <input type="checkbox"/>	I feel confident that encryption makes online services safe for me to use.
Yes <input type="checkbox"/> No <input type="checkbox"/>	I am concerned about understanding my rights as described by the terms of the application provider.
<input type="checkbox"/> Representative	Is the stakeholder a real existing entity? Is this stakeholder used as proxy for a group of homogeneous stakeholders?
<input type="checkbox"/> Group	Is the stakeholder not a real existing entity? Is this stakeholder used to describe a group of homogeneous stakeholders?
<input type="checkbox"/> Role	Can this stakeholder be shared through groups of heterogeneous stakeholders? Are there well-defined rights for this stakeholder?
Trustworthiness goals for addressing trust concerns of the end-user are listed as below:	
Trustworthiness goals	<input type="checkbox"/> Safety <input type="checkbox"/> Security <input type="checkbox"/> Reliability <input type="checkbox"/> Privacy <input type="checkbox"/> Adaptability <input type="checkbox"/> Testability <input type="checkbox"/> Compliance
	<input type="checkbox"/> Cost <input type="checkbox"/> Usability <input type="checkbox"/> Compatibility <input type="checkbox"/> Performance <input type="checkbox"/> Maintainability <input type="checkbox"/> Configurability
	<input type="checkbox"/> Complexity <input type="checkbox"/> Data-related quality <input type="checkbox"/> Correctness <input type="checkbox"/> Resolution <input type="checkbox"/> Predictability
	<input type="checkbox"/> Accessibility of content and data

Trust worthiness requirements pattern

The trustworthiness requirements pattern serves as a structured approach to define and express trust-related needs in a system. It relies on linguistic templates that provide a clear and consistent way of capturing requirements such as reliability, security, privacy, and transparency. By following this pattern, stakeholders can ensure that trustworthiness aspects are not only identified but also articulated in a manner that reduces ambiguity. This approach supports effective communication among developers, analysts, and decision-makers, making it easier to align system design with the expectations of users and regulatory standards.

Context:

This pattern supports the requirement engineer in establishing a clear connection between trustworthiness requirements and trustworthiness goals. In addition, it associates the identified requirements with the corresponding

trustworthiness properties that must be integrated within the business process model, ensuring that the overall trustworthiness objectives are effectively achieved.

Problem and forces:

The primary issue addressed by this pattern solution is the lack of alignment between business process models and goal models. This inconsistency creates difficulties in ensuring that organizational goals are properly reflected in process execution, leading to gaps in performance and effectiveness.

Solution:

This approach for defining trustworthiness requirements applies linguistic templates to describe both the requirements themselves and the related process elements and resources within a business process. The method of specifying trustworthiness requirements and further refining them into detailed trustworthiness properties is presented in Table 2.

TW.Req.<ID>	<Descriptive name for the trustworthiness requirement>
Author Name	<Name of the author>
Process Element	<Description on the process element and its descriptive name>
Business Process	<Business process that the process element is related to, (process name and process ID)>
Goal of the Process	<Operational goals that the process supposes to address, (goal and the ID of the goal model)>
Trustworthiness Goal	<The objective goal toward enabling trustworthiness which relates to this process element (trustworthiness goal name, ID of goal model)>
Trustworthiness properties	<List of trustworthiness properties, which the process elements should be capable of>
Metrics	The metrics which can be applied in order to quantify the trustworthiness properties and calculate the metric values are: <list of metrics>
Thresholds	The metric values must: (<metric value> > <threshold value> <metric value> < <threshold value>) (<metric value> = <desired value> <upper bound threshold value> > <metric value> > <lower bound threshold value>) <upper bound threshold value> => <metric value> => <lower bound threshold value>) Metric values fulfil the following constraint: <constraint>
Target	Lower values are desired Higher values are desired
Source	<source on which metrics can be applied or metric values can be obtained, e.g., log files, sensors>
Responsible	role department organisation person
Informed	role department organisation person
Comments	<Additional information about the trustworthiness, e.g., condition assumptions, and criteria for the process elements such as activities or resource, etc.>



TABLE 2: Consequences:

The primary outcome of implementing the suggested patterns is the establishment of well-

structured and traceable trustworthiness requirements, ensuring sufficient uniformity and reliability.

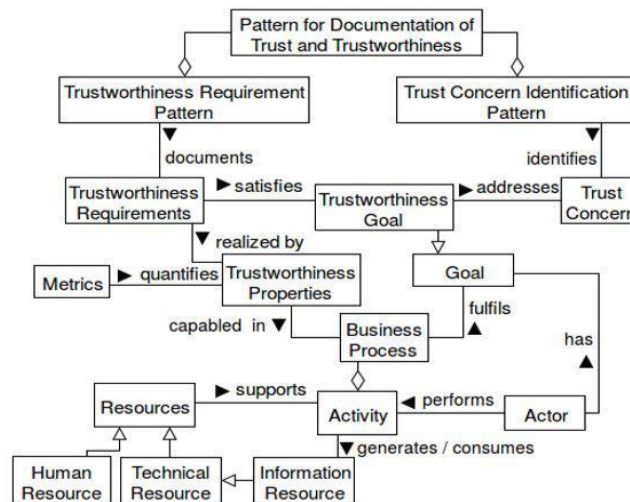


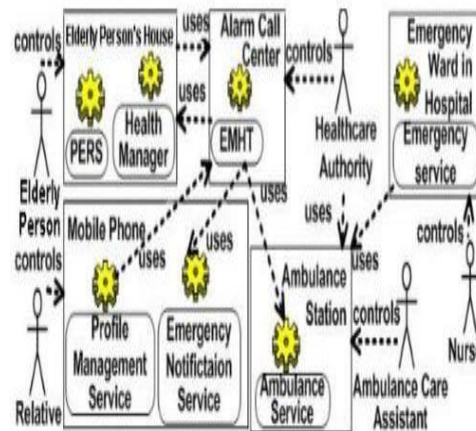
Fig:3 The structure of the patterns and the relation of the concepts

Example

A middle-aged woman who lives with her children often feels anxious after experiencing a fall in the bathroom when no one was around to help her. She remained on the floor for quite some time before managing to call for assistance. Later, she

came to know about some AAL services designed to provide safety and quick support in such emergencies. She is now thinking about adopting one of these services to ensure her well-being and reduce the risk of facing similar difficulties again.

Patterns for Identification of Trust Concerns and Specification of Trustworthiness Requirements



The primary stakeholders in the healthcare sector can be categorized into three groups. (1) End-users, specifically the elderly population, as they are the main recipients and direct beneficiaries of healthcare services. (2) Technology providers, which include organizations offering medical applications such as elderly monitoring systems, fall detection devices, telecommunication services, and internet connectivity essential for digital health solutions. (3) Healthcare service providers, encompassing healthcare authorities, medical centers, clinics, and hospitals that are responsible for delivering medical care and ensuring the well-being of patients.

Evaluation

In this study, we explored various patterns and examined their interconnections to identify and

refine system requirements, while also comparing our findings with those of related research. To achieve the research objectives, we reviewed several primary studies in the field of software development; however, we found limited work specifically addressing simple and comprehensive questions for end-users. Our focus was on recognizing and refining end-user trust concerns and analyzing how these concerns influence requirements. The results of our analysis are presented in Table 1. Relevant studies were retrieved from databases using the following search strings:

Pattern approach

- Identification of non-functional requirements in MSDA
- Refinement of requirements in MSDA
- Elicitation of requirements in MSDA

Results	Identification and refinement of requirements by using pattern based approach	Patterns for identification of trust concerns and specification of trustworthiness requirements	A Framework for Object Identification and Refinement Process in Object-Oriented Analysis and Design	A Process for Pattern Identification, Authoring, and Application	A Framework for Identifying and Analyzing Non-functional Requirements from Text
Easy to Understand	✓	-	✓	-	✓
Promote reuse	✓	✓	-	-	✓
More comprehensive	✓	-	-	✓	-
Reduce ambiguities	✓	✓	✓	-	-

TABLE 1
CONCLUSION

In this paper, we created a supportive and collaborative environment by applying patterns and their interconnections, and by designing simple, clear, and user-friendly questions for the end-users. We also focused on identifying and refining the requirements of the system model in line with the mental models of users, thereby resolving their difficulties in understanding the system. As system requirements continue to grow rapidly, it is estimated that the cognitive load of end-users will multiply significantly, while modern software becomes more widely integrated into daily life to provide convenience. However, not all IT solutions are secure, as only around 60% of modern applications undergo proper security evaluation and testing, which exposes them to risks. To counter such threats, specialized approaches such as authentication mechanisms, secure boot processes, data encryption during communication, and timely alerts for possible attacks should be implemented. Furthermore, system models should be strengthened to prevent denial-of-service vulnerabilities, and users should consider updating or restoring their software specifications to more secure versions, ensuring protection against modern software risks and unauthorized access.

Acknowledgment

I would like to sincerely express my deep gratitude to Dr. Mudasil Khan for his insightful and constructive guidance throughout the planning and development of this research work. His generous willingness to dedicate his time and expertise has been immensely appreciated. I would also like to thank Dr. Mudasil Khan, instructor of Research Methodology, for his significant contribution to this research, which greatly helped in the successful completion of this paper.

REFERENCES

- [1] Patterns for Identification of Trust Concerns and Specification of Trustworthiness Requirements (July 2016) Nazila Gol Mohammadi and Maritta Heiselpaluno - The Ruhr Institute for Software Technology, University of Duisburg-Essen, Germany
- [2] A Framework for Object Identification and Refinement Process in Object-Oriented Analysis and Design. Romi S. Wahono Department of Information and Computer Sciences Saitama University, Behrouz H. Far Department of Electrical and Computer Engineering, University of Calgary
- [3] A Process for Pattern Identification, Authoring, and Application Christoph Fehling, Johanna Barzen, Uwe Breitenbücher, Frank Leymann

- Institute of Architecture of Application Systems University of Stuttgart , Germany
- [4] A Framework for Identifying and Analyzing Non-functional Requirements from Text VibhuSaujanya Sharma, Roshni R. Ramnani, Shubhashis Sengupta Accenture Technology Labs Accenture, Bangalore, India
- [5] A Catalogue of Functional Software Requirement Patterns for the Domain of Content Management Systems C. Palomares, C. Quer, X. Franch Universitat Politècnica de Catalunya (UPC) Barcelona, Catalunya, Spain, S. Renault, C. Guerlain Public Research Centre Henri Tudor Luxembourg, Luxembourg
- [6] RFP-Cog: Linguistic-based Identification and Mapping of Service Requirements in Request for Proposals (RFPs) to IT Service Solutions Hamid R. Motahari-Nezhad, Juan M. Cappi, Taiga Nakamura, Mu Qiao IBM Almaden Research Center San Jose, CA, United States (2016).
- [7] Ashfa Umer, Imran Sarwar Bajwa, M. Asif Naeem, NL-Based Automated Software Requirements Elicitation and Specification, Advances in Computing and Communications in Computer and Information Science Volume 191, Springer. 2011, pp 30-39.
- [8] A Process for Pattern Identification, Authoring, and Application Christoph Fehling, Johanna Barzen, Uwe Breitenbcher, Frank Leymann Institute of Architecture of Application Systems University of Stuttgart , Germany (2017).
- [9] Towards a Model Driven Refinement Process through Architecture Evaluation Grzegorz Loniewski^{1,2}, Etienne Borde¹, Emilio Insfran²
¹Institut Telecom; TELECOM ParisTech; LTCI - UMR 5141 46, rue Barrault, 75013, Paris, France (2013).
- [10] S. Sen and N. Stojanovic. Gruve: A methodology for complex event pattern life cycle management. In 22nd International Conference on Advanced Information Systems Engineering (CAiSE10), Hammamet , Tunisa, June 09-11-2010, Proceedings, Lecture Notes in Computer Science. Springer, 2010.
- [11] J barzen ,u breitenbcher : they provide the solution of language easing pattern composition in different domains through architecture Evaluation 2020.
- [12] M Kleehaus, C Caprano Systematic Literature literature through a backward search of the analyzed papers in the literature search process Inparallel, we will perform a structured survey among companies in Review of model driven Refinement process Scalability for Large Scale Projects 2018