The AgileCo Pattern Language: Physical Environment

Daniel Pinho

Faculty of Engineering, University of Porto
Porto, Portugal
up201505302@fe.up.pt

Ademar Aguiar

Faculty of Engineering, University of Porto
INESC TEC
Porto, Portugal
ademar.aguiar@fe.up.pt

ABSTRACT

Communication is crucial in agile contexts due to the need for collaboration and the number of different people involved, ranging from developers to users and business people. This communication should be effective to ensure everyone is on the same page; otherwise it can lead to undesired results such as misunderstanding and loss of productivity. As such, there is a need to ensure good communication in agile contexts. This paper presents a pattern language to tackle this issue, alongside with the patterns of a section of the language. The pattern language aims to provide a set of practices that help with recurrent problems and increase the effectiveness of communication. The patterns presented in this paper (INFORMATION RADIATORS, CO-LOCATED TEAM, OPEN SPACE, INFORMAL COMMUNICATION SPACE and CUBES) address the issues related with the physical environment surrounding an agile team.

CCS CONCEPTS

• **Social and professional topics** → *Project and people management*; • **Software and its engineering** → *Agile software development*; *Collaboration in software development*; • **General and reference** → General conference proceedings.

KEYWORDS

agile software development, communication, collaboration, knowledge sharing, physical environment

ACM Reference Format:

Daniel Pinho and Ademar Aguiar. 2020. The AgileCo Pattern Language: Physical Environment. In *European Conference on Pattern Languages of Programs 2020 (EuroPLoP '20)*, July 1–4, 2020, Virtual Event, Germany. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3424771.3424790>

1 INTRODUCTION

In the software engineering world, one can find several ways and paradigms to develop software. One of those ways involves using agile methods. Agile methodologies started to appear in the 1990s, based off of lean manufacturing principles, and value people, adaptability, having customers involved in the development process and delivering working software. These values are present in

the Agile Manifesto [3], a document signed by supporters of agile methodologies that outlines the basics of such models.

Communication is an important point of interest to the good functioning of agile methodologies, being mentioned several times in the Agile Manifesto [3]. This communication is done between developers and stakeholders, who have vastly different realities and world views. Effective communication is described as the type of communication where the recipient understands the entirety of the message's meaning and can return said meaning back to the speaker [14]. The differences between the realities of the people involved in agile contexts make effective communication highly important, as its absence can cause misunderstanding, waste of time and loss of productivity. In order to mitigate these unwanted effects, it is important to strive towards good communication in agile contexts.

This paper presents the AgileCo (**A**gile **E**ffective **C**ommunication, pronounced "Agile Echo") pattern language, which aims to improve communication in agile contexts and increase its effectiveness, mitigating the unwanted effects mentioned previously. The pattern language is targeted at agile coaches, who have a support role in helping development teams and stakeholders with the application of agile methodologies. The rest of this paper includes an overview of the AgileCo pattern language, the set of patterns related to the physical environment where agile teams and other personnel work together, and a conclusion.

2 BACKGROUND AND RELATED WORK

Agile software development methodologies often approach concepts related to communication, as shown by their texts: the Scrum Guide [20] showcases examples of effective meetings, the importance of feedback and how the people involved should work as a team; when using XP, developers are encouraged to practise pair programming, which ensures a steady stream of verbal communication and feedback, as well as increasing trust within the team [2][6]. This practice also influences the working spaces, as XP encourages the usage of shared, open offices [2]. Scaled agile methodologies tend to address the topic of communication more explicitly due to the increased number of people working; for example, Large Scale Scrum (LeSS) [11] expresses the importance of both verbal and virtual communication, the self-organisation of teams, and that these should be co-located.

The literature also explores the importance of communication in agile contexts, as well as how the environment surrounding the people involved influences their performance. Several papers showcase the importance of information radiators and how they are useful in transmitting information and reducing communication overhead [6][15][18][21]. In the digital world, collaborative spaces

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

EuroPLoP '20, July 1–4, 2020, Virtual Event, Germany

© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-7769-0/20/07...\$15.00

<https://doi.org/10.1145/3424771.3424790>

such as wikis and intranets also play important roles in diffusing information throughout the organisation [5][12].

Open space brings both advantages and disadvantages: on the one hand, it reduces physical distances between team members and their communication costs, facilitates informal communication, fosters trust within the team and makes it easier to employ pair programming practices [6][16][18][21]. On the other hand, open space offices may become noisy and distracting, potentially reducing productivity [18]. One can address these issues through the use of half-height glass walls, as they create privacy without creating too much distance between team members [16]. All these factors influence the convection currents of information in the working space, as information flows more freely when people are close together and there are fewer barriers between them [6].

The differences between co-located and distributed teams are other topics of note: team members who work in the same location tend to have more trust in each other, as well as having better feedback loops due to instant, informal communication [12]. Meanwhile, distributed teams, while having greater geographical freedom, must be more mindful of regular team building activities to maintain the unity within the team [6].

There are also related pattern languages that, while not being written through the lens of communication in agile settings, approach these topics: Coplien and Harrison's *Organizational Patterns of Agile Software Development* [8] explore concepts related to people's roles, work distribution, information flow and the organisational environment, among others. The Scrum Patterns [19] document practices and concepts related to Scrum, including teamwork, the organisation and useful behaviours and practices.

3 PATTERN LANGUAGE OVERVIEW

The pattern language here presented contains a set of interconnected patterns with the aim of helping people within agile environments to identify existing problems and hindrances and solving them. They will make it possible to increase the effectiveness of the communication within these environments. The patterns documented here were mined from existing literature and documentation for existing agile methodologies.

The resulting pattern language (see Figure 1) provides a set of good practices and other factors that improve communication as a whole in agile teams and organisations, reducing side effects from ineffective communication and making knowledge transfer easier.

The pattern mining process deviated slightly from traditional methods such as the usage of interviews. Instead, the bulk of the information documented in these patterns originates from a literature review of relevant materials, such as books, web pages and research papers. This literature review was complemented by some empirical knowledge in a few, select cases.

The patterns' structure is in accordance with Christopher Alexander's form of *name-context-problem-solution* [1] accompanied by a set of forces, which influence the solution for the problem, accompanying them. The patterns are divided into three areas: people patterns, which are related to how individuals interact in the organisation and their skill-set; environment patterns, related to the physical and digital spaces where organisation members work; and

activity patterns, which approach the meetings and practices people engage in during their day-to-day work.

Environment patterns

These patterns relate to the factors, both physical and digital, that exist in the environment where people are situated and, therefore, influence how they behave. The patlets of these patterns can be found in Table 1 in Appendix A.

Physical. These patterns express tangible, physical elements that enhance communication in agile contexts, attempting to remove barriers and reduce the face-to-face communication cost.

Digital. The digital environment also influences how people communicate in agile contexts as different communication methods have different goals and yield different results.

Activity patterns

These patterns relate to the activities people engage in during their day-to-day activities, which can be divided into meetings and practices. The patlets of these patterns can be found in Table 2 in Appendix A.

Meetings. Meetings are recurring events where people gather to discuss topics relevant to the organisation's activities. Each meeting has a specific goal in mind.

Practices. Practices are activities that are done during the daily activity of the organisation, with various forms and objectives.

People patterns

These patterns relate to people and how they behave as individuals within the organisation, which has some sort of structure. Individuals often have a role within the environment and should possess a set of skills adequate for communicating effectively. The patlets of these patterns can be found in Table 3 in Appendix A.

Roles. An individual has a designated role within the organisation. These roles have clearly defined responsibilities and established interactions between themselves.

Organisational structure. An organisation needs to have some sort of structure so that its day-to-day activities can happen as they should. These patterns explain several organisation types, as well as detailing some additional characteristics about the interpersonal relationships between personnel.

Skills. Organisation members should possess skills that can help in communicating with others; these skills can be related to their behaviour and how they view the world or related to their interactions with others.

PATTERN: INFORMATION RADIATORS

In an agile development setting, team members are often sitting in close proximity to each other, sharing an OPEN SPACE office or simply being a CO-LOCATED TEAM. In these environments, the team members are working on the same project, towards the same goal.

Figure 1: An overview of the pattern language with the patterns organised into categories and subcategories.

Problem

A team cannot achieve its goal without some sort of coordination between its members. The usage of meetings during each iteration, particularly DAILY MEETINGS, can improve the coordination existent between the team members. However, this information is only provided once a day, being static, and it relies on the team members' memory.

How can we ensure everyone is up to speed regarding the current status of the project?

Forces

- **Energy requirements.** Team members should not have to spend much of their energy, both physical and mental, to stay up to date on the project status.
- **Accessibility.** The information being conveyed to the team members should be easily accessible to them.
- **Information stickiness.** Having things written down ensures they are not lost; Team members should have the option to consult the information available to them if necessary [6].
- **Information overload.** Having access to large amounts of information all at once can be detrimental to our ability to

process them; too much information starts acting like noise and can cause errors, stress, distractions and overall lower quality of information [10].

Solution

Include visual aids, such as whiteboards, flipcharts and similar items with useful information in common areas or in places where people frequently walk by.

These visual aids are often displayed on walls of hallways and shared offices, ensuring accessibility by most, if not all, of the DEVELOPMENT TEAM and the project stakeholders [6].

Depending on the people that use it, the size of an information radiator may vary: two or three people can use a sheet of paper while a whole team should use something bigger. However, one should strive to aim for a bigger size as it is more visible and thus easier to take in information [6].

Another important thing is that the information displayed on an information radiator changes over time; this can be used to track the current progress during an iteration, look up task assignments or even check on the status of auxiliary services [6].

Related patterns

The usage of information radiators to keep track of the progress made so far during an iteration is useful when it comes to GOAL SETTING. They can also provide the outcomes of RETROSPECTIVE MEETINGS, so that team members know what to keep in mind to improve their performance [6]. Information radiators may also take a non-physical form, as digital ones can be part of OPEN INFORMATION TOOLSETS.

Consequences

Information radiators allow for the receiving of important information in a relatively short amount of time. However, having up-to-date information requires a conscious effort to update information radiators, and it is possible to find outdated information in one of them. Additionally, the information present in the radiators should be focused on a particular topic; not doing this may lead to cases of information overload by the people who use the radiators.

PATTERN: CO-LOCATED TEAM

In agile software development projects, the members of the DEVELOPMENT TEAM interact and communicate regularly with each other to ensure they are properly coordinating their efforts.

Problem

The communication between team members, which is inherent to the way teams work, should be easy to achieve. However, people may tend to not talk to each other if they find resistance or if they find that it takes too much effort.

How can we ensure that communication is nearly effortless within the team?

Forces

- **Communication costs.** The distance between team members ends up affecting the cost of communication between them. A team member that needs to ask something to a coworker who is working on another floor, they will spend time and energy going to meet their colleague. If the coworker happens to be away from their office, all this effort will have been in vain, and the team member will not be as motivated to try again [6].
- **Conway's law.** Melvin Conway [7] states that a design's structure will be a reflection of the organisation's communication structure. As such, if an organisation wants to have a good, modular design, the interactions between team members should be able to happen accordingly [9].
- **Effectiveness of informal communication.** Informal communication is significant in agile projects to the point that one may not notice that it is there, but its absence makes that importance apparent. Team members being able to communicate face-to-face informally during the day allows them to build trust with each other, discuss any unplanned issues that may appear and absorb tacit knowledge [17] from each other, improving the overall familiarity with the project.
- **Interruptions.** Building software is a process that requires concentration, as developer combines their thoughts and ideas to create a cohesive set of instructions [6]. Interruptions break this concentration and can result in errors or additional development time. People working in environments that are more prone to interruptions may tend to be less productive.

Solution

The team members should work in close proximity to each other, in the same room, in adjacent rooms or at least on the same floor.

© Daniel Pinho

Out of all these cases, being in the same room is the most desirable option, as it is the option with the lowest distance between team members. However, if such is not possible, working in adjacent rooms or on the same floor can be a worthwhile alternative [6].

Having team members work in close proximity allows them to clear up with the relevant person any issues that may be on their mind in person, build their working relationships and to develop higher-quality code.

The proximity also reduces any potential frustration and subsequent loss of motivation due to missed encounters, as there is a high probability that the people in question will run into each other frequently [6].

This pattern is a stark contrast to having distributed teams, in which team members work separated across different regions.

Related patterns

The team may evolve their proximity regarding working in the same room, developing an OPEN SPACE layout to their offices.

Consequences

The nature of co-located work implies team members will be working in an office. Team members have to commute to work which, compared to remote work, has its own set of advantages and disadvantages [13][4].

Interruptions are an additional issue of working in the same location; while a person can mitigate potential interruptions and distractions that may come from their digital tools, being interrupted because someone knocked on their door or just walked up to them is not something that is easily avoided.

PATTERN: OPEN SPACE

Agile teams work as cohesive units, making use of their highly frequent communication and teamwork to deliver working software in small intervals of time (i.e. iterations).

Problem

The physical space where team members work is of high importance to the resulting work, as the latter is conditioned by the cooperation between team members. This cooperation should be as smooth as possible with the least amount of friction; if the workplace does not foster a collaborative atmosphere, the team will face problems with reaching its goal.

How should a CO-LOCATED TEAM organise its space to reduce friction and entropy?

Forces

- **Distance between members.** The motivation team members have to talk to each other is directly related to the distance between them and the energy (both physical and mental) spent to perform that interaction.
- **Feeling at ease.** Team members are bound to engage in conversation with their colleagues regarding work and other subjects depending on how comfortable and familiar they are with each other and the amount of trust within the team.
- **Face-to-face communication.** Face-to-face communication is an effective communication mechanism that comes naturally to human beings. It is able to convey a lot more than just words, showcasing feelings and other sub-textual information in a quick interaction.

- **Osmotic communication.** Alistair Cockburn [6] states that just by being in another person's vicinity, one can receive information about current events and other topics simply by hearing things passively, lowering the cost of communication to a certain degree.

Solution

The team should organise their furniture to remove physical barriers between team members, including everyone in the working atmosphere.

This requires that the team is working in the same room [6], and the furniture layout should not exclude anyone, allowing people to see each other and to help team members feel more at ease in their colleagues' presence [18][16].

An open space layout enables team members to have ad-hoc conversations with their colleagues, where they can ask for help, guidance or any other sort of assistance required, which allows for better knowledge sharing and can make the group more cohesive.

Consequences

Despite the more inclusive atmosphere that an open space office provides, team members can feel overwhelmed and distracted due to their apparent constant availability and exposure to the environment [18].

This can be mitigated by the addition of quieter areas where they can relax and have some privacy and alone time [15], and by making use of INFORMAL COMMUNICATION SPACES.

In addition, the noise that may result from an open space office creates an environment that may not be beneficial for calls to happen. As such, an organization that employs open space offices may want to invest in CUBES to ensure some quieter areas.

PATTERN: INFORMAL COMMUNICATION SPACE

Agile software development is a collaborative process. Since team members and stakeholders are often working in a CO-LOCATED fashion, it is normal that face-to-face communication, a highly useful means of transmitting information, is also commonplace in these environments.

Problem

Despite the usefulness that face-to-face communication enjoys in agile contexts, it is important that it does not take over the entirety

of the development team's time. As such, one must be aware that there is a time and a place for everything.

What is the best environment for informal communication?

Forces

- **Focus.** One can describe software development as being akin to building a tower. A developer must tie ideas together, ensuring that the resulting code is coherent and that its logic is sound. This process requires a lot of concentration, and this tower falls apart if the developer loses focus [6], which can happen if they are suddenly roped in a conversation, which is not advantageous to the project.
- **Encouraging interactions.** Informal communication should be encouraged within the organisation because it fosters TRUST and knowledge sharing within the project personnel, and not stifled in order to ensure the developers spend their whole workday developing code.

Solution

Areas that are far enough from workstations to avoid disturbing people who are working but close enough where people are in the vicinity are often good spaces to dedicate to informal communication.

© Daniel Pinho

The distance from workstations makes it so that people engaging in conversations can avoid disturbing any team members who might be busy coding. The area in question can also have some INFORMATION RADIATORS, such as whiteboards, to support any conversations happening there.

This area should also be accessible to everyone and, perhaps, a place where people often walk by or that is close enough to the DEVELOPMENT TEAM's workstations. This ensures that if a team member needs to talk to a colleague, there is a high chance they will stumble upon them in that space during the day.

It is important to recognise, however, that not every organisation has the available room or resources to structure the layout of its working spaces in a way that is favourable to the existence of an INFORMAL COMMUNICATION SPACE. As such, this pattern should be seen as a luxury to have and not a vital necessity.

Related patterns

A SNACK SHRINE [19] and THE WATER COOLER [8] are good examples of INFORMAL COMMUNICATION SPACES, as they provide central gathering places around which informal conversations can happen.

PATTERN: CUBES

The software development world is getting more and more connected across borders and continents. As such, it is not uncommon to find situations where project members have to participate in meetings using CALLS.

Problem

When participating in a conference call, good acoustic conditions are a requirement, so that all parties can understand what is being said. If the call includes a video feed, then that should be of quality as well.

How can we create good conditions for conference calls?

Forces

- **Subject sensitivity.** Meeting participants may want to avoid outside parties listening in on meetings that are dealing with sensitive or confidential subjects.
- **Background noise.** The place where the call participants are in should be as quiet as possible to maximise everyone's understanding of the discussion taking place.
- **Hardware.** Depending on how conference calls are set up in the organisation, specific hardware may be necessary, even if it is just a screen and a camera for video conferences.

Solution

Using acoustically-sound rooms or spaces, that can fit either a couple of people or a large group, provide good conditions for calls and meetings.

© Daniel Pinho

These acoustically isolated rooms (CUBES) can be fruitful to ensure that project members can participate in conference calls without having to worry about noise, which can be distracting and may lead to communication issues.

If a large group of people (such as a DEVELOPMENT TEAM) is participating in a call, a meeting room may be suitable for them. However, for individuals or pairs, one can go with commercially available "phone booths" or with small rooms.

It is worth noting that an organisation may not have the space or the resources available for these CUBES. As such, they should be treated as something nice to have but that is not vital for the normal, day-to-day activities of the DEVELOPMENT TEAM.

Examples

There are several companies offering soundproof booths as commercial solutions to enterprises, such as ROOM, Zenbooth and Urban Office.

Related patterns

CUBES may be a result of an OPEN SPACE office and, as mentioned earlier, they provide a support role to CALLS. However, if a CUBE is large enough for a small group of people, they can also be useful rooms for meetings such as RETROSPECTIVE MEETINGS.

4 CONCLUSIONS

Agile software development methods are one of the many ways one can develop software. These methodologies involve the participation of many different people with very different roles and backgrounds and, as such, communication must be as effective as it can be to reduce potential problems.

This paper presents a pattern language that aims to provide proven practices and solutions that foster effective communication in different areas: the environment surrounding project personnel, the activities team members partake in, and the people themselves. This paper also presents, in further detail, the pattern language's five patterns that relate to the physical environment: INFORMATION RADIATORS, CO-LOCATED TEAMS, OPEN SPACE, INFORMAL COMMUNICATION SPACE, and CUBES.

The authors aim to continue the development of this pattern language, streamlining some parts and researching others even more.

ACKNOWLEDGMENTS

To Elissaveta Gourova, for her valuable guidance during the shepherding process. To Waheedullah Khail, Michael Krisper, Ömer Uldağ, and Valentino Vranić for their feedback and insights during the writer's workshop.

REFERENCES

- [1] Christopher Alexander, Sara Ishikawa, and Silverstein Murray. 1977. *A Pattern Language*. Oxford University Press, New York, New York, USA.
- [2] Kent Beck and Cynthia Andres. 2004. *Extreme Programming Explained: Embrace Change* (2 ed.). Addison-Wesley. 224 pages.
- [3] Kent Beck, Mike Beedle, A Van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, and Dave Thomas. 2001. Principles behind the Agile Manifesto. <http://agilemanifesto.org/principles.html>
- [4] Raleigh Burgess. 2001. Systems and methods for providing remote support via productivity centers. US Patent App. 09/813,663.
- [5] Thomas Chau and Frank Maurer. 2004. Knowledge sharing in agile software teams. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 3075 (2004), 173–183. https://doi.org/10.1007/978-3-540-25967-1_12
- [6] Alistair Cockburn. 2006. *Agile software development: the cooperative game*. Addison-Wesley.
- [7] Melvin E. Conway. 1968. How do committees invent. *Datamation* 14, 4 (1968), 28–31.
- [8] James O Coplien and Neil B Harrison. 2004. *Organizational Patterns of Agile Software Development*. Prentice Hall.
- [9] James D. Herbsleb and Rebecca E. Grinter. 1999. Architectures, coordination, and distance: Conway's law and beyond. *IEEE Software* 16, 5 (1999), 63–70. <https://doi.org/10.1109/52.795103>
- [10] Orrin Edgar Klapp. 1986. *Overload and boredom: Essays on the quality of life in the information society*. Greenwood Publishing Group Inc.
- [11] Craig Larman and Bas Vodde. 2016. *Large-scale Scrum: More with LeSS*. Addison-Wesley.
- [12] Lucas Layman, Laurie Williams, Daniela Damian, and Hynek Bures. 2006. Essential communication practices for Extreme Programming in a global software development team. *Information and Software Technology* 48, 9 (2006), 781–794. <https://doi.org/10.1016/j.infsof.2006.01.004>
- [13] Liang Ma and Runing Ye. 2019. Does daily commuting behavior matter to employee productivity? *Journal of Transport Geography* 76, February (apr 2019), 130–141. <https://doi.org/10.1016/j.jtrangeo.2019.03.008>
- [14] Perry McIntosh, Richard Luecke, and Jeffery H. Davis. 2008. *Interpersonal Communication Skills in the Workplace, Second Edition*. AMACOM Div American Mgmt Assn. <http://books.google.com/books?id=PETL2Ai1HkUC&pgis=1>
- [15] Deepti Mishra and Alok Mishra. 2009. Effective communication, collaboration, and coordination in eXtreme programming: Human-centric perspective in a small organization. *Human Factors and Ergonomics In Manufacturing* 19, 5 (sep 2009), 438–456. <https://doi.org/10.1002/hfm.20164>
- [16] Orlando Murru, Roberto Deias, and Giampiero Mugheddu. 2003. Assessing XP at a European Internet company. *IEEE Software* 20, 3 (2003), 37–43. <https://doi.org/10.1109/MS.2003.1196318>
- [17] Ikujiro Nonaka and Noboru Konno. 1998. The Concept of “Ba”: Building a Foundation for Knowledge Creation. *California Management Review* 40, 3 (apr 1998), 40–54. <https://doi.org/10.2307/41165942>
- [18] Minna Pikkariainen, Jukka Haikara, Outi Salo, Pekka Abrahamsson, and Jari Still. 2008. The impact of agile practices on communication in software development. *Empirical Software Engineering* 13, 3 (jun 2008), 303–337. <https://doi.org/10.1007/s10664-008-9065-9>
- [19] Jeff Sutherland, James O Coplien, Lachlan Heasman, Mark den Hollander, and Cesário Ramos. [n.d.]. *Scrum Patterns*. <http://scrumbook.org/>
- [20] Jeff Sutherland and Ken Schwaber. 2017. *The Scrum Guide*. (2017).
- [21] Don Wells. 2013. *Extreme Programming*. <http://www.extremeprogramming.org/>

A PATLETS

This appendix provides the tables that include patterns' patlets. Environment pattern patlets can be found in Table 1, Activity pattern patlets in Table 2, and People pattern patlets in Table 3.

Table 1: Environment pattern patlets

Physical	
Pattern	Patlet
INFORMATION RADIATORS	Visual aids such as whiteboards, flipcharts and similar items with useful information help ensuring everyone is up to speed when placed in common areas or in places where people frequently walk by.
CO-LOCATED TEAM	Having team members work in close proximity to each other, in the same room, adjacent rooms or at least on the same floor helps communication to be nearly effortless within the team.
OPEN SPACE	Organising the team's furniture to remove physical barriers between team members and include everyone in the working atmosphere helps reducing friction and entropy.
INFORMAL COMMUNICATION SPACE	Areas far enough from workstations but close enough where people are in the vicinity often are good spaces to dedicate to informal communication.
CUBES	Acoustically-sound rooms or spaces are places that create good conditions for conference calls.
Digital	
Pattern	Patlet
CALLS	A group of people who are far apart from each other use a phone, a computer or any similar device to participate in a call, allowing them to interact synchronously.
INSTANT MESSAGING	Using a computer or another internet-accessible device, a person can send short text messages that are sent instantly to their recipient through the use of an IM service.
E-MAILS	E-mails can be used to address team members and provide them with pondered, substantial information that can be archived and referred to later on.
OPEN INFORMATION TOOLSET	Web-based tools where users can easily search for and add new information provide a good platform for information archival and knowledge sharing.

Table 2: Activity pattern patlets

Meetings	
Pattern	Patlet
DAILY MEETINGS	The Development Team can stay updated about the project by meeting at the start of each day and discussing their progress on the project, their current tasks and any problems they might be facing.
PLANNING MEETINGS	Before an iteration starts, the Development Team and the Customers should decide which tasks should be done during the iteration, establishing a clear course of action.
REVIEW MEETINGS	At the end of an iteration, the Development Team should meet with the Customers and other stakeholders to present the increment built during said iteration, evaluating it and reflecting on the work done.
RETROSPECTIVE MEETINGS	In order to grow, the Development Team should meet with the Coach between the end of an iteration and the beginning of the next one somewhere quiet and reflect on the team's work and on how to improve it.
Practices	
Pattern	Patlet
PAIR PROGRAMMING	In order to make knowledge distribution easier, development team members should pair up and work together at the same workstation, while one codes and the other provides support.
BACKLOG REFINEMENT	The team's adaptability is aided by periodically assessing the backlog, modifying existing items depending on their relevance, and adding new ones if necessary.
COLLABORATIVE DECISION MAKING	Decisions should be made through the use of collaborative decision-making methods, such as finding consensus, trying to reach a compromise, or applying a voting method, in order to make decisions.

Table 3: People pattern patlets

Roles	
Pattern	Patlet
COACH	The Coach supports the development team and the management in the adoption and learning process of agile methodologies, along with the removal of obstacles, and facilitation of meetings.
DEVELOPMENT TEAM	The developers should organise themselves into teams, entities where people work together and support each others' activities, share leadership, hold each other accountable, share a strong bond, and work towards a common goal.
CUSTOMERS	The Customers should have a close working relationship with the Development Team, providing feedback and clearing up any questions they may have.
MANAGEMENT	When dealing with agile projects, the management should support their teams by forecasting needs and removing both internal and external obstacles.
Organisational structure	
Pattern	Patlet
AD-HOC ORGANISATION	The organisation members should feel free to interact with any other person if the need arises, regardless of rank or area of operation.
COMMUNITIES	Communities are groups that exist parallel to the organisational chart, allowing people to exchange knowledge between teams, increase inter-team collaboration and build a sense of unity within the organisation.
LOW-DEPTH STRUCTURE	The organisational chart should be structured in order to have a low number of layers, encouraging a more direct approach to communication and decision-making in the organisation.
Skills	
Pattern	Patlet
TRUST	Building trusting relationships between team members allows them to feel more comfortable around each other and to support each other's weaknesses, permitting them to complement each other.
HUMILITY	Team members should be humble and think about others as equals instead of holding themselves in higher regard over their peers, even when feeling accomplished.
EFFECTIVE LISTENING	Effective listening techniques, such as reducing distractions, expressing interest, asking clarifying questions and paraphrasing are useful to maximise information retained from face-to-face communication.
NEW IDEAS	Generating new ideas be done through techniques such as dialogue, brainstorming, Nominal Group Technique (NGT) or the Six Thinking Hats.
COMMITMENT	Even in the most difficult of situations, team members should establish clear courses of action and buy into them.
ACCOUNTABILITY	Team members should hold each other responsible for their work by establishing high standards and not being able to give feedback, regardless of how hard it may be.
FEEDBACK	Team members should provide feedback to each other, by pointing out each other's strong points, as well as any potential improvement points that can be corrected.
GOAL SETTING	Dividing a project into SMART goals can ensure the resulting objectives are specific, time-bound and have clear end conditions.