
Communication via Design Artifacts in Software Development Teams

Adriana Lopes Damian¹, Clarisse Sieckenius de Souza², Tayana Conte¹

¹ **USES Research Group**

Federal University of Amazonas Manaus, AM – Brazil
{adriana, tayana}@icomp.ufam.edu.br

² **Semiotic Engineering Research Group**

PUC-Rio
Rio de Janeiro, Brazil
clarisse@inf.puc-rio.br



USES Technical Report
RT-USES-2020-0011
October, 2020

Institute of Computing (IComp)
Federal University of Amazonas (UFAM)
Manaus, Amazonas 69077-000

ABSTRACT

In this technical report, we present Directives of Communicability (DCs) adapted for the production of specific artifacts, such as UML use case and diagram, and prototype. We also present DCs adapted to support practitioners in the diagnosis of some software artifacts, such as UML use case, and prototype. Besides, we describe factors that support practitioners to analyze the conditions of communication via artifacts. Practitioners from five different development teams used our proposal. In this technical report, we present our data analysis of the factors used by practitioners.

1. DIRECTIVES OF COMMUNICABILITY

We proposed the DCs for supporting producers to improve the artifacts' content to promote the communication through them, i.e. the communicability of software artifacts, especially software design artifacts (SDA) [1]. Each directive instructs producers to reflect on essential characteristics of effective communication. These characteristics, according to the Semiotic Engineering, are based on the Grice's Cooperative Principle, which is established from four Grice maxims [2], as follows:

Quality - Try to make your contribution one that is true;

Quantity - Make your contribution as informative as is required. Do not make your contribution more informative than is required;

Relation – Be relevant; and

Manner - Be perspicuous. Avoid obscurity of expression. Avoid ambiguity.

We describe the proposed directives, based on the Grice Maxims [2], below. DCs can be applied in different software artifacts that represent solutions designed for systems development. Producers must employ each of the directives sequentially. Further details of the DCs proposal and steps for your application are available in [1].

“Say the truth!” - DC1: Use true information. Do not use information that affects the quality of the artifact (maxim of Quality).

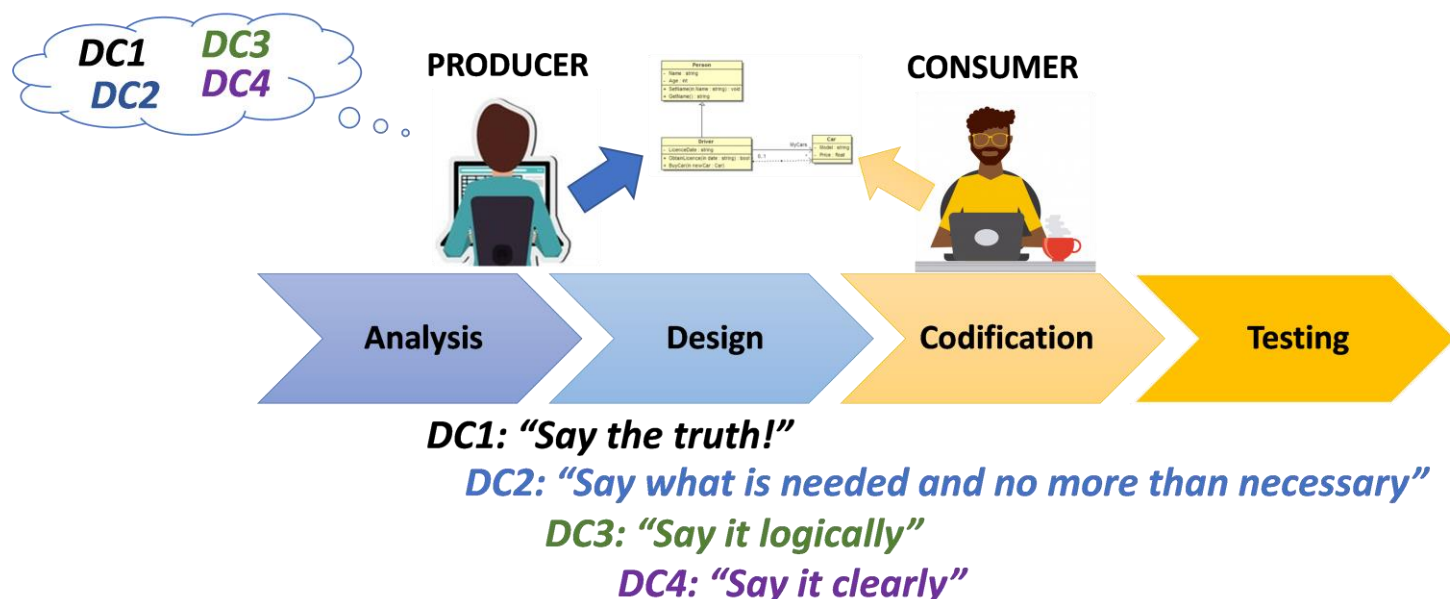
“Say what is needed and no more than necessary” - DC2: Use the necessary content in the template. Do not use unnecessary content in the model (maxim of Quantity).

“Say it logically” - DC3: Organize the information in the model consistently (maxim of Relation).

“Say it clearly” - DC4: Organize the information in the model clearly (maxim of Manner).

Regarding the application of DCs in SDA, for instance, we present DCs specifics for UML class diagrams, and prototype in the next pages in this technical report. After this, we present the DCs adapted to support practitioners in the diagnosis of UML use case and prototype.

DIRECTIVES OF COMMUNICABILITY



UML Class Diagrams

DC1: "Say the truth!" - Use true information. Do not use information that affects the quality of the artifact.

- Are there classes that are not part of the problem domain?
- Do not insert outdated information

DC2: "Say what is needed and no more than necessary": Use the necessary content in the template. Do not use unnecessary content in the model

- Are all necessary classes of the problem domain in the diagram?
- Insert all necessary relationships
- Do not insert unnecessary content in the diagram

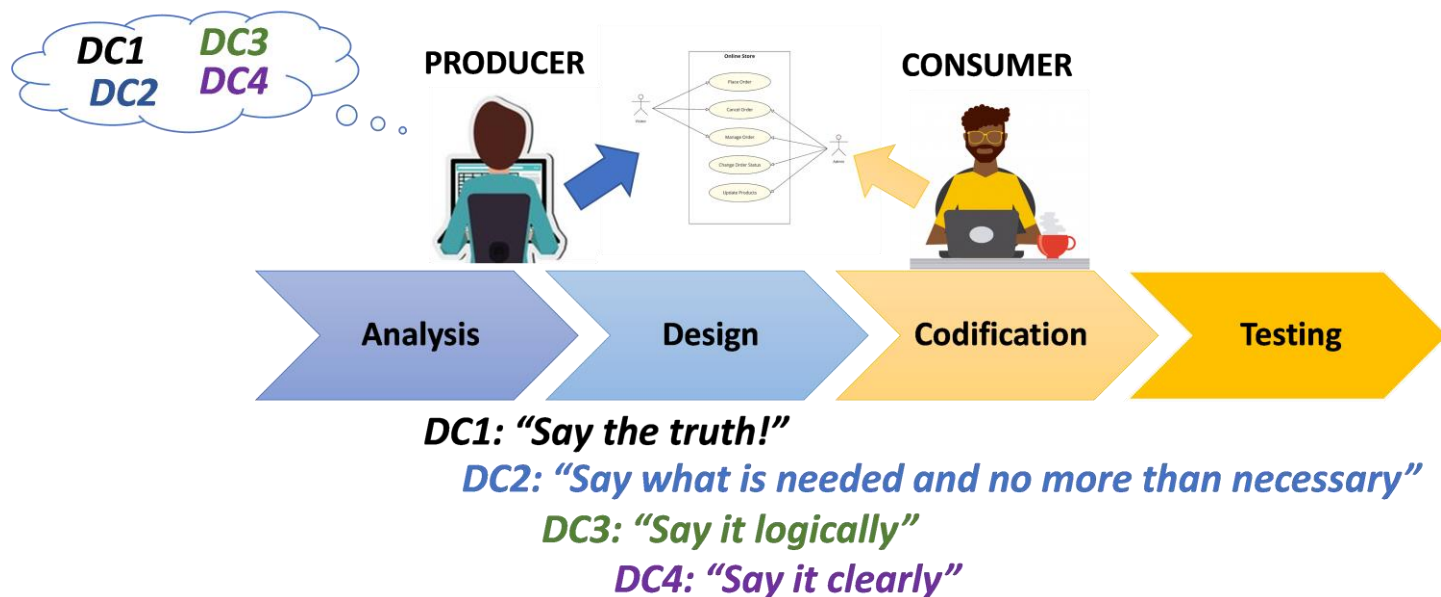
DC3: "Say it logically" - Organize the information in the model consistently

- Organize classes so that they are best viewed
- Organize relationships to represent a better view of the classes

DC4: "Say it clearly" - Organize the information in the model clearly

- Are there classes and relationships with descriptions that are not clear?
- Avoid ambiguity and implicit interpretation

DIRECTIVES OF COMMUNICABILITY



UML Use Case

DC1: "Say the truth!" - Use true information. Do not use information that affects the quality of the artifact.

- Are there uses cases that are not part of the problem domain?
- Do not insert outdated information

DC2: "Say what is needed and no more than necessary": Use the necessary content in the template. Do not use unnecessary content in the model

- Are all necessary use cases are there in the diagram?
- Insert all necessary relationships
- Do not use unnecessary content in the diagram
- Analyze the amount of information in the specification of all use cases

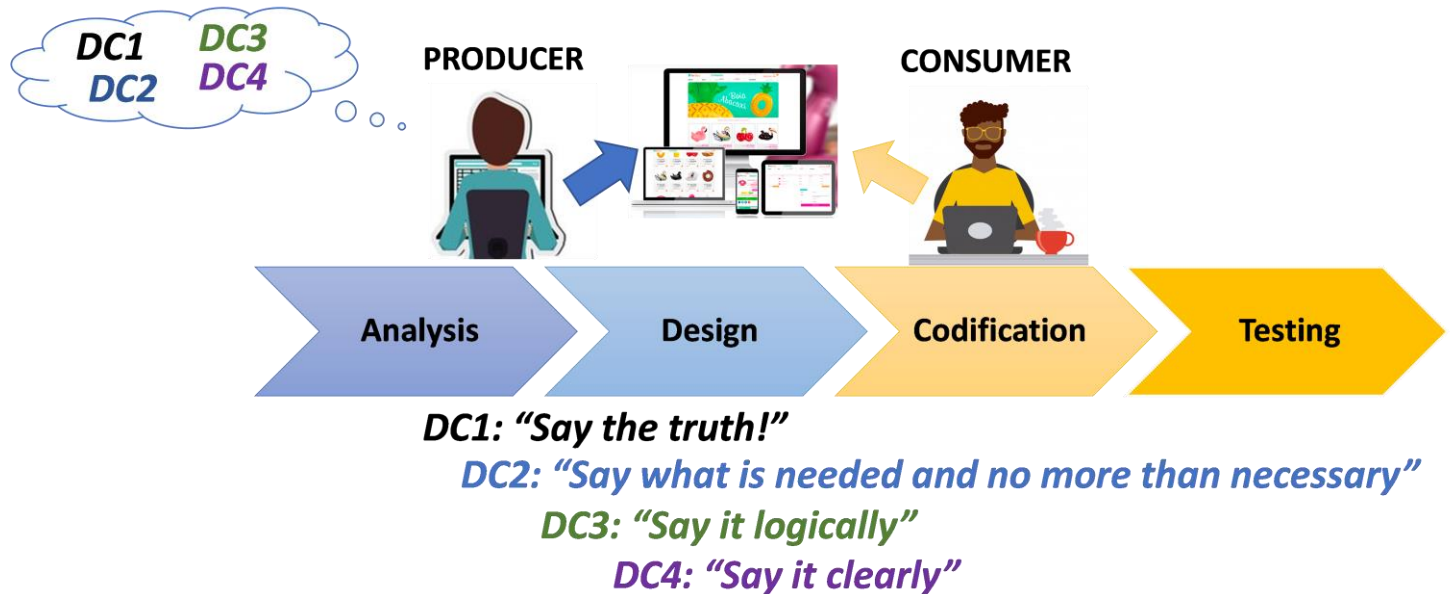
DC3: "Say it logically" - Organize the information in the model consistently

- Organize use cases in the diagram so that they are best viewed
- Organize relationships to represent a better view of the use cases in the diagram
- Organize the use cases in the specification so that they present a logical sequence for the consumers

DC4: "Say it clearly" - Organize the information in the model clearly

- Describe the names of the use cases so that they are easily understood and differentiated from each other
- Avoid ambiguity and implicit interpretation

DIRECTIVES OF COMMUNICABILITY



Prototypes

DC1: "Say the truth!" - Use true information. Do not use information that affects the quality of the artifact.

- Analyze if there is outdated information in the prototypes or that are outside the problem domain
- Do not insert outdated information

DC2: "Say what is needed and no more than necessary": Use the necessary content in the template. Do not use unnecessary content in the model

- Analyze whether the number of screens, with their respective interface elements, is sufficient for consumers to comprehend the system
- Do not use unnecessary content in the diagram

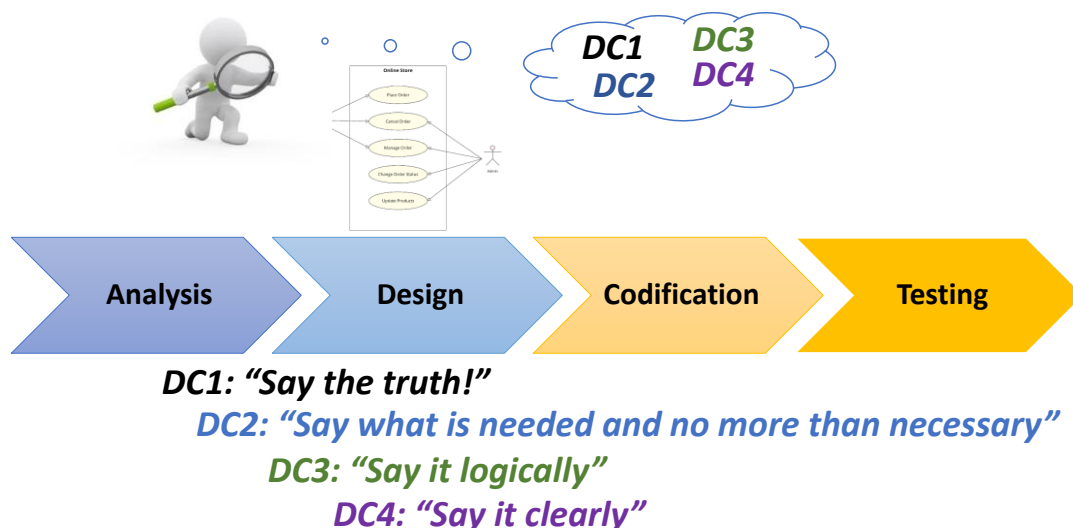
DC3: "Say it logically" - Organize the information in the model consistently

- Organize the screen in the specification so that they present a logical sequence for the consumers

DC4: "Say it clearly" - Organize the information in the model clearly

- Use terms in the content of the screens that can be easily comprehended and differentiated from each other
- Avoid ambiguity and implicit interpretation

Communicability Problems Diagnosis with DIRECTIVES OF COMMUNICABILITY



UML Use Cases

DC1: "Say the truth!" - Does the information that is not true in the use cases?

- o Is there information that is not part of the problem domain? If so, information that is not part of the system may have been codified.
- o Is there inconsistent information? If so, this may have caused the coding of inconsistent/incorrect features.
- o Is there outdated information? If so, this may have caused the codification of features not necessary/incorrect.

DC2: "Say what is needed and no more than necessary" - Necessary information, and not more than necessary, is in the use case?

- o Are all necessary information being represented in the use cases? If not, this may have caused functionalities to be omitted from the system.
- o Is there unnecessary information for the team to understand? If so, this may have caused ambiguity in understanding the prototype, leading to the codification of inconsistent features due to multiple interpretations.

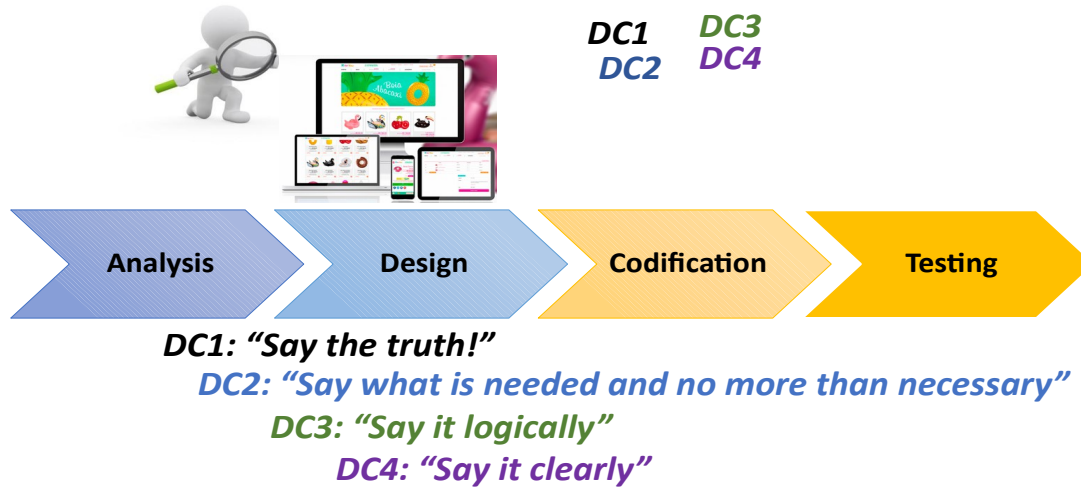
DC3: "Say it logically" - Is the information organized sequentially? Is this organization relevant to the team's understanding?

- o Is the screen sequence disorganized? If so, this probably made it difficult to understand the information in the prototypes, causing the development of non-prioritized/inconsistent features.
- o Is the organization of the screens relevant to the order of development of the screens? If not, this likely caused the development of parts of the system that are not prioritized in the system.

DC4: "Say it clearly" - Is there information difficult to understand?

- o Are there concepts implicit in these use cases? If so, this is likely to make it difficult for new members during system maintenance. In addition, this may have caused functionality to be omitted from the system.
- o Is it easy to understand the solution in uses cases? If not, this probably caused different inconsistencies in the system's behavior.

Communicability Problems Diagnosis with DIRECTIVES OF COMMUNICABILITY



Prototypes

DC1: "Say the truth!" - Does the information that is not true in the prototypes?

- o Is there information that is not part of the problem domain? If so, information that is not part of the system may have been codified.
- o Is there inconsistent information? If so, this may have caused the coding of inconsistent/incorrect features.
- o Is there outdated information? If so, this may have caused the codification of features not necessary/incorrect.

DC2: "Say what is needed and no more than necessary" - Necessary information, and not more than necessary, is in the prototypes?

- o Are all necessary information being represented in the prototypes? If not, this may have caused functionalities to be omitted from the system.
- o Is there unnecessary information for the team to understand? If so, this may have caused ambiguity in understanding the prototype, leading to the codification of inconsistent features due to multiple interpretations.

DC3: "Say it logically" - Is the information organized sequentially? Is this organization relevant to the team's understanding?

- o Is the screen sequence disorganized? If so, this probably made it difficult to understand the information in the prototypes, causing the development of non-prioritized/inconsistent features.
- o Is the organization of the screens relevant to the order of development of the screens? If not,

DC4: "Say it clearly" - Is there information difficult to understand?

- o Are there concepts implicit in these prototypes? If so, this is likely to make it difficult for new members during system maintenance. In addition, this may have caused functionality to be omitted from the system.
- o Is it easy to understand the solution in prototypes? If not, this probably caused different inconsistencies in the system's behavior.

2. FACTORS PROPOSED FOR THE ANALYSIS OF COMMUNICATION CONDITIONS VIA ARTIFACT

We proposed factors that help analyze the conditions of communication via artifact, providing better support to practitioners interested in this type of analysis. We elaborated these factors based on Semiotic Engineering [4][5] and in the communication space model proposed by Jakobson [3]. The factors are listed below, and the proposed items to measure each of the factors are described in Table1.

Artifact as a Means of Communication - which refers to understanding whether the artifact can clearly represent the problem domain, and whether the artifacts, as it is made available, can support the team in their activities.

Communicative Content - which refers to understanding whether the artifact can establish reciprocal communication between producers and consumers.

Artifact Consumers - Regarding this last factor, we can divide it into *consumer's ability* to know about the artifact, i.e. whether s/he is able to understand the artifacts' notation, and whether the consumer considers such *artifact useful* to support him in his activities.

TABLE I. FACTORS PROPOSED FOR THE ANALYSIS OF COMMUNICATION CONDITIONS VIA ARTIFACT

Factors	Items
Artifact as Means of Communication	<ul style="list-style-type: none"> •M1. I believe that this artifact is necessary to support the team in understanding the problem domain. •M2. This artifact has elements in its notation to represent the best way to communicate with the team about the problem domain. •M3. This artifact has elements in its notation that I intend to use to talk to the team about, for instance, the documentation or information for implementing the software. •M4. This artifact assists me in what I would like to “convey” to the team about mastering the problem. •M5. This artifact allows my project for the software to be represented clearly for the team. •M6. This is the most suitable artifact for the team to understand this perspective of the software I am creating (interaction, structural, behavioral, etc.). •M7. This artifact is sufficient to represent such a perspective for software development.
Communicative Content	<ul style="list-style-type: none"> •Cont1. I believe that the modeling elements I am using are understandable to the team. •Cont2. No disorganized information was introduced in the artifact. •Cont3. Excessive information has not been introduced into the artifact. •Cont4. The amount of information in the artifact is sufficient for the team to understand. •Cont5. I believe that the content of the artifact is straightforward and easy to understand.
Artifact Consumers	<ul style="list-style-type: none"> •PC1. I can recognize the informational content represented in this artifact. •PC2. I find it easy to understand this artifact. •PC3. I will adopt this artifact to understand my development activities. •PC4. I will understand the software better with this artifact. •PC5. I find it easy to use this artifact in my activities.

These factors were applied by practitioners from five different teams to understand the conditions of communication via artifacts. The items were answered in the following sequence by artifacts producers and consumers:

PRODUCERS

- **Cont4.** The amount of information in the artifact is sufficient for the team to understand.
- **M6.** This is the most suitable artifact for the team to understand this perspective of the software I am creating (interaction, structural, behavioral, etc.).
- **M2.** This artifact has elements in its notation to represent the best way to communicate with the team about the problem domain.
- **M5.** This artifact allows my project for the software to be represented clearly for the team.
- **M1.** I believe that this artifact is necessary to support the team in understanding the problem domain.
- **M7.** This artifact is sufficient to represent such a perspective for software development.
- **M3.** This artifact has elements in its notation that I intend to use to talk to the team about, for instance, the documentation or information for implementing the software.
- **M4.** This artifact assists me in what I would like to “convey” to the team about mastering the problem.
- **Cont1.** I believe that the modeling elements I am using are understandable to the team.
- **Cont2.** No disorganized information was introduced in the artifact.
- **Cont5.** I believe that the content of the artifact is straightforward and easy to understand.
- **Cont3.** Excessive information has not been introduced into the artifact.

CONSUMERS

- **PC1.** I can recognize the informational content represented in this artifact.
- **PC5.** I find it easy to use this artifact in my activities.
- **PC3.** I will adopt this artifact to understand my development activities.
- **PC2.** I find it easy to understand this artifact.
- **PC4.** I will understand the software better with this artifact.

The first author of this work sent online questionnaires to the different members of each team. About the results on the application of DCs by practitioners, we present the results in the next sections.

3.1 Analysis Results of Communication Conditions

ARTIFACT AS MEANS OF COMMUNICATION (ANSWERED BY PRODUCER ARTIFACT)	strongly agree	agree	somewhat agree	neutral	somewhat disagree	disagree	strongly disagree
M1. I believe that this artifact is necessary to support the team in understanding the problem domain.		1					
M2. This artifact has elements in its notation to represent the best way to communicate with the team about the problem domain.	1						
M3. This artifact has elements in its notation that I intend to use to talk to the team about, for instance, the documentation or information for implementing the software.			1				
M4. This artifact assists me in what I would like to “convey” to the team about mastering the problem.	1						
M5. This artifact allows my project for the software to be represented clearly for the team.	1						
M6. This is the most suitable artifact for the team to understand this perspective of the software I am creating (interaction, structural, behavioral, etc.).		1					
M7. This artifact is sufficient to represent such a perspective for software development.		1					
COMMUNICATIVE CONTENT (ANSWERED BY PRODUCER ARTIFACT)							
Cont1. I believe that the modeling elements I am using are understandable to the team.		1					
Cont2. No disorganized information was introduced in the artifact.		1					
Cont3. Excessive information has not been introduced into the artifact.		1					
Cont4. The amount of information in the artifact is sufficient for the team to understand.	1						
Cont5. I believe that the content of the artifact is straightforward and easy to understand.	1						
ARTIFACT CONSUMERS (ANSWERED BY CONSUMER ARTIFACT)							
PC1. I can recognize the informational content represented in this artifact.	1	2					
PC2. I find it easy to understand this artifact.	1	1	1				
PC3. I will adopt this artifact to understand development activities	1	2					
PC4. I will understand the software better with this artifact.		2	1				
PC5. I find it easy to use this artifact.	2	1					

Figure 1: Factors Related to Communication via Artifact in the Team 1.

ARTIFACT AS MEANS OF COMMUNICATION (ANSWERED BY PRODUCER ARTIFACT)	strongly agree	agree	somewhat agree	neutral	somewhat disagree	disagree	strongly disagree
M1. I believe that this artifact is necessary to support the team in understanding the problem domain.		2					
M2. This artifact has elements in its notation to represent the best way to communicate with the team about the problem domain.		2					
M3. This artifact has elements in its notation that I intend to use to talk to the team about, for instance, the documentation or information for implementing the software.		2					
M4. This artifact assists me in what I would like to “convey” to the team about mastering the problem.		2					
M5. This artifact allows my project for the software to be represented clearly for the team.	2						
M6. This is the most suitable artifact for the team to understand this perspective of the software I am creating (interaction, structural, behavioral, etc.).	2	1					
M7. This artifact is sufficient to represent such a perspective for software development.		1	1				
COMMUNICATIVE CONTENT (ANSWERED BY PRODUCER ARTIFACT)							
Cont1. I believe that the modeling elements I am using are understandable to the team.		2					
Cont2. No disorganized information was introduced in the artifact.	1				1		
Cont3. Excessive information has not been introduced into the artifact.	1	1					
Cont4. The amount of information in the artifact is sufficient for the team to understand.	1		1				
Cont5. I believe that the content of the artifact is straightforward and easy to understand.		2					
ARTIFACT CONSUMERS (ANSWERED BY CONSUMER ARTIFACT)							
PC1. I can recognize the informational content represented in this artifact.	1	2					
PC2. I find it easy to understand this artifact.	1	1	1				
PC3. I will adopt this artifact to understand development activities	1	2					
PC4. I will understand the software better with this artifact.		2	1				
PC5. I find it easy to use this artifact.	2	1					

Figure 2: Factors Related to Communication via Artifact in the Team 2.

ARTIFACT AS MEANS OF COMMUNICATION (ANSWERED BY PRODUCER ARTIFACT)	strongly agree	agree	somewhat agree	neutral	somewhat disagree	disagree	strongly disagree
M1. I believe that this artifact is necessary to support the team in understanding the problem domain.	1						
M2. This artifact has elements in its notation to represent the best way to communicate with the team about the problem domain.	1						
M3. This artifact has elements in its notation that I intend to use to talk to the team about, for instance, the documentation or information for implementing the software.	1						
M4. This artifact assists me in what I would like to “convey” to the team about mastering the problem.	1						
M5. This artifact allows my project for the software to be represented clearly for the team.	1						
M6. This is the most suitable artifact for the team to understand this perspective of the software I am creating (interaction, structural, behavioral, etc.).	1						
M7. This artifact is sufficient to represent such a perspective for software development.		1					
COMMUNICATIVE CONTENT (ANSWERED BY PRODUCER ARTIFACT)							
Cont1. I believe that the modeling elements I am using are understandable to the team.	1						
Cont2. No disorganized information was introduced in the artifact.	1						
Cont3. Excessive information has not been introduced into the artifact.	1						
Cont4. The amount of information in the artifact is sufficient for the team to understand.		1					
Cont5. I believe that the content of the artifact is straightforward and easy to understand.	1						
ARTIFACT CONSUMERS (ANSWERED BY CONSUMER ARTIFACT)							
PC1. I can recognize the informational content represented in this artifact.	2	2					
PC2. I find it easy to understand this artifact.	1	2	1				
PC3. I will adopt this artifact to understand development activities	2	2					
PC4. I will understand the software better with this artifact.	2	2					
PC5. I find it easy to use this artifact.	1	3					

Figure 3: Factors Related to Communication via Artifact in the Team 3.

ARTIFACT AS MEANS OF COMMUNICATION (ANSWERED BY PRODUCER ARTIFACT)	strongly agree	agree	somewhat agree	neutral	somewhat disagree	disagree	strongly disagree
M1. I believe that this artifact is necessary to support the team in understanding the problem domain.	1						
M2. This artifact has elements in its notation to represent the best way to communicate with the team about the problem domain.	1						
M3. This artifact has elements in its notation that I intend to use to talk to the team about, for instance, the documentation or information for implementing the software.	1						
M4. This artifact assists me in what I would like to “convey” to the team about mastering the problem.	1						
M5. This artifact allows my project for the software to be represented clearly for the team.	1						
M6. This is the most suitable artifact for the team to understand this perspective of the software I am creating (interaction, structural, behavioral, etc.).	1						
M7. This artifact is sufficient to represent such a perspective for software development.	1						
COMMUNICATIVE CONTENT (ANSWERED BY PRODUCER ARTIFACT)							
Cont1. I believe that the modeling elements I am using are understandable to the team.	1						
Cont2. No disorganized information was introduced in the artifact.	1						
Cont3. Excessive information has not been introduced into the artifact.	1						
Cont4. The amount of information in the artifact is sufficient for the team to understand.	1						
Cont5. I believe that the content of the artifact is straightforward and easy to understand.	1						
ARTIFACT CONSUMERS (ANSWERED BY CONSUMER ARTIFACT)							
PC1. I can recognize the informational content represented in this artifact.	2	3					
PC2. I find it easy to understand this artifact.	2	2	1				
PC3. I will adopt this artifact to understand development activities	2	3					
PC4. I will understand the software better with this artifact.	2	2	1				
PC5. I find it easy to use this artifact.	3	2					

Figure 4: Factors Related to Communication via Artifact in the Team 4.

ARTIFACT AS MEANS OF COMMUNICATION (ANSWERED BY PRODUCER ARTIFACT)	strongly agree	agree	somewhat agree	neutral	somewhat disagree	disagree	strongly disagree
M1. I believe that this artifact is necessary to support the team in understanding the problem domain.		1				1	
M2. This artifact has elements in its notation to represent the best way to communicate with the team about the problem domain.	1		1				1
M3. This artifact has elements in its notation that I intend to use to talk to the team about, for instance, the documentation or information for implementing the software.		1				1	
M4. This artifact assists me in what I would like to “convey” to the team about mastering the problem.		1				1	
M5. This artifact allows my project for the software to be represented clearly for the team.		1				1	
M6. This is the most suitable artifact for the team to understand this perspective of the software I am creating (interaction, structural, behavioral, etc.).		1		1			
M7. This artifact is sufficient to represent such a perspective for software development.		1		1			
COMMUNICATIVE CONTENT (ANSWERED BY PRODUCER ARTIFACT)							
Cont1. I believe that the modeling elements I am using are understandable to the team.	1	1					
Cont2. No disorganized information was introduced in the artifact.	1	1					
Cont3. Excessive information has not been introduced into the artifact.	1	1					
Cont4. The amount of information in the artifact is sufficient for the team to understand.	1	1					
Cont5. I believe that the content of the artifact is straightforward and easy to understand.		2					
ARTIFACT CONSUMERS (ANSWERED BY CONSUMER ARTIFACT)							
PC1. I can recognize the informational content represented in this artifact.	5	6	1				
PC2. I find it easy to understand this artifact.	4	8					
PC3. I will adopt this artifact to understand development activities	4	2		3	2	1	
PC4. I will understand the software better with this artifact.	4	4		1	1	2	
PC5. I find it easy to use this artifact.	6		1	1	3	1	

Figure 5: Factors Related to Communication via Artifact in the Team 5.

3.2 Evaluation of the Items proposed for each Factor

Regarding the evaluation of the items proposed for each factor, we carried out an exploratory analysis to verify whether these items are considered reliable and whether they are related. We carried out this evaluation on a set of data obtained after applying the questionnaire, and analyzed with the support of the SPSS tool (v.23). For this analysis in the SPSS tool, we organized the responses of the practitioners as follows:

strongly agree – 1/ **agree** – 2/ **somewhat agree** – 3/ **neutral** – 4/ **somewhat disagree** -5/ **disagree** -6/ **strongly disagree** – 7

	M1	M2	M3	M4	M5	M6	M7
PRODUCER 1 – Software Designer of Team 1	3,00	1,00	3,00	1,00	3,00	1,00	1,00
PRODUCER 2 – Designer of Team 2	1,00	1,00	3,00	1,00	3,00	1,00	1,00
PRODUCER 3 – Requirements Analyst of Team 2	2,00	2,00	2,00	1,00	2,00	2,00	2,00
PRODUCER 4 – Software Designer of Team 3	2,00	1,00	1,00	1,00	1,00	1,00	2,00
PRODUCER 5 – Software Designer of Team 4	1,00	1,00	1,00	1,00	1,00	1,00	1,00
PRODUCER 6 - Software Designer of Team 5	2,00	1,00	2,00	2,00	2,00	2,00	2,00
PRODUCER 7 - Requirements Analyst of Team 5	5,00	3,00	5,00	5,00	5,00	3,00	3,00

	CONT1	CONT2	CONT3	CONT4	CONT5
PRODUCER 1 – Software Designer of Team 1	1,00	1,00	3,00	1,00	3,00
PRODUCER 2 – Designer of Team 2	5,00	5,00	1,00	3,00	5,00
PRODUCER 3 – Requirements Analyst of Team 2	2,00	2,00	2,00	1,00	1,00
PRODUCER 4 – Software Designer of Team 3	2,00	1,00	1,00	2,00	1,00
PRODUCER 5 – Software Designer of Team 4	1,00	1,00	1,00	1,00	1,00
PRODUCER 6 - Software Designer of Team 5	2,00	2,00	2,00	2,00	2,00
PRODUCER 7 - Requirements Analyst of Team 5	1,00	1,00	1,00	1,00	2,00

	PC1	PC2	PC3	PC4	PC5
CONSUMER 1 – Developer 1 of Team 1	3,00	3,00	3,00	1,00	2,00
CONSUMER 2 - Developer 2 of Team 1	2,00	2,00	2,00	1,00	2,00
CONSUMER 3 - Developer 3 of Team 1	2,00	1,00	2,00	3,00	1,00
CONSUMER 4 - Developer 1 of Team 2	2,00	2,00	1,00	1,00	2,00
CONSUMER 5 - Developer 2 of Team 2	1,00	1,00	1,00	1,00	1,00
CONSUMER 6 - Developer 3 of Team 2	2,00	1,00	2,00	1,00	1,00
CONSUMER 7 - Developer 1 of Team 3	1,00	1,00	2,00	2,00	2,00
CONSUMER 8 - Developer 2 of Team 3	2,00	3,00	2,00	2,00	2,00
CONSUMER 9 - Developer 3 of Team 3	1,00	2,00	1,00	1,00	2,00
CONSUMER 10 - Developer 4 of Team 3	2,00	2,00	1,00	1,00	1,00
CONSUMER 11 - Developer 1 of Team 4	2,00	2,00	2,00	1,00	2,00
CONSUMER 12 - Developer 2 of Team 4	2,00	3,00	2,00	3,00	2,00
CONSUMER 13 - Developer 3 of Team 4	2,00	2,00	2,00	2,00	1,00
CONSUMER 14 - Developer 4 of Team 4	1,00	1,00	1,00	2,00	1,00
CONSUMER 15 - Developer 5 of Team 4	1,00	1,00	1,00	1,00	1,00
CONSUMER 16 - Developer 1 of Team 5	2,00	2,00	1,00	1,00	1,00
CONSUMER 17 - Developer 2 of Team 5	1,00	1,00	1,00	1,00	1,00
CONSUMER 18 - Developer 3 of Team 5	2,00	2,00	4,00	1,00	1,00
CONSUMER 19 - Developer 4 of Team 5	2,00	2,00	1,00	2,00	1,00
CONSUMER 20 - Developer 5 of Team 5	2,00	2,00	1,00	2,00	1,00
CONSUMER 21 - Developer 6 of Team 5	1,00	1,00	4,00	5,00	4,00
CONSUMER 22 - Developer 7 of Team 5	1,00	1,00	4,00	5,00	4,00
CONSUMER 23 - Developer 9 of Team 5	1,00	2,00	2,00	2,00	1,00
CONSUMER 24 - Developer 9 of Team 5	2,00	2,00	3,00	2,00	4,00
CONSUMER 25 - Developer 10 of Team 5	3,00	2,00	5,00	4,00	6,00
CONSUMER 26 - Developer 11 of Team 5	2,00	2,00	2,00	1,00	3,00
CONSUMER 27 - Developer 12 of Team 5	1,00	1,00	3,00	3,00	4,00

We applied Cronbach's Alpha test to analyze the reliability of the items. The results showed a high internal consistency for 'Artifact as Means of Communication' ($p = 0.876$), 'Communicative Content' ($p = 0.940$), and 'Artifact Consumers' ($p = 0.700$). We conducted a factor analysis to verify whether the items are related. It is observed in Table 2 that all items of 'Artifact as Means of Communication' were related to Component 1. The same was achieved for 'Communicative Content' and Component 2. Both indicates that the items are related with these factors.

TABLE II. FACTOR ANALYSIS: 'ARTIFACT AS MEANS OF COMMUNICATION' AND 'COMMUNICATIVE CONTENT'.

	COMPONENT 1	COMPONENT 2
M1	0,892	-0,229
M2	0,935	-0,148
M3	0,919	0,258
M4	0,756	-0,337
M5	0,889	-0,320
M6	0,919	0,258
M7	0,875	-0,243
CONT1	-0,179	0,939
CONT2	-0,093	0,951
CONT3	-0,241	0,845
CONT4	0,116	0,910
CONT5	-0,132	0,963

Regarding the factor analysis for 'Artifact Consumers', the results show that the items are related to the two sub-factors, as shown in Table 3. We emphasize that this is the only factor that is measured by two consumer's perspectives: consumer's ability to know about the artifact, and artifact's usefulness to support him in his activities.

TABLE III. FACTOR ANALYSIS: 'CONSUMERS OF ARTIFACT'

	COMPONENT 1	COMPONENT 1
PC1	0,916	0,160
PC2	0,901	-0,038
PC3	0,167	0,915
PC4	-0,254	0,838
PC5	0,083	0,907

References

- [1] A. Lopes, E. Oliveira, T. Conte and C. S. de Souza, “Directives of Communicability: towards better communication through software models”, Proceedings of the 12th International Workshop on Cooperative and Human Aspects of Software Engineering, 2019, pp. 45-48.
- [2] H. P. Grice, “Logic and Conversation”, Syntax and Semantics 3: Speech arts, ed. Peter Cole and Jerry Morgan, 1975, pp. 41–58.
- [3] R. Jakobson, “Linguistics and poetics”, T. A. Sebeok (ed.), Style in Language. Cambridge, MA: The MIT Press, 1960, pp. 350–377.
- [4] C. S. De Souza, “The Semiotic Engineering of Human-Computer Interaction (Acting with Technology)” The MIT Press, (1st ed.), 2005.
- [5] C. S. de Souza, R. F. de G. Cerqueira, L. M. Afonso, R. R. de M. Brandão and J. S. J. Ferreira, “Software Developers as Users: Semiotic Investigations in Human-Centered Software Development”, (1st ed.), Springer International Publishing Switzerland, 2016.