

Storyboarding as a Means of Requirements Elicitation and User Interface Design: An Application to the Drones' Industry



Ramón Mollá, Virginia Santamarina-Campos, Francisco Abad,
and Giovanni Tipantuña

Abstract The goal of this chapter is to present the best practices and usage of storyboarding during the initial definition of multidisciplinary projects where partners from different backgrounds (engineering, arts, creative industries, etc.) collaborate to define the main user tasks to be implemented during the project. One of the challenges in this phase of the project is to be able to effectively communicate the ideas between the partners. Every background has his own concepts, technical language and procedures, and sometimes it is hard to convey in words the real meaning of an idea. It is even possible that different disciplines use the same tools, but have different names and different purposes. Storyboards are universally understandable and provide a common ground for sharing ideas and for discussing and discovering new points of view.

1 Introduction

Creative Industries (CIs) have used storyboarding for many years. It is a very useful and powerful tool for describing the content of a linear production, such as an animation film (Finch 2011). Storyboarding is a good collaborative technique, where all the members of the group can internalise the whole project and see the whole picture as well as small details. It also enables anyone to contribute his or her ideas effectively. Homogeneous teams of animators or audiovisual professionals regularly use this tool. Also, multidisciplinary multimedia production teams can work concurrently (Taylor 2013).

R. Mollá (✉) · F. Abad · G. Tipantuña
Instituto Universitario de Automática e Informática Industrial (ai2), Universitat Politècnica de València, Valencia, Spain
e-mail: rmolla@dsic.upv.es

V. Santamarina-Campos
Conserv. & Restoration of Cult. Heritage Department, Universitat Politècnica de València, Valencia, Spain

It is very common nowadays in the software development industry to design interactive systems based on user-centred design (UCD) processes. Any mid-to-large-sized modern software project typically involves multidisciplinary teams composed of technical and not technical stakeholders: end-users, clients, producers, designers and engineers. The different and complementary perspectives of the team members define the requirements of the system to be constructed. The ISO standard 24765:2010(E) defines requirement as “a condition or capability that must be met or possessed by a system, system component, product, or service to satisfy an agreement, standard, specification, or other formally imposed documents” (ISO/IEEE 24765-2010(E)). The software requirements specification document is the result of the requirement analysis stage of the project, and must consider all aspects of the application, ranging from user needs to non-functional requirements, such as safety, security, performance, reliability or latency requirements.

The methodology to obtain clear user requirements is key for the success of the project. This methodology has to:

1. Get all team members involved in the early stages of the system design.
2. Allow for the effective communication of the design ideas among all members.
3. Document the decisions in a way that all team members can understand and use them in an appropriate way in later stages of the process.

Understanding user requirements is an integral part of information systems design and it is critical to the success of interactive systems. However, specifying these requirements is not so simple to achieve. As specified in the ISO 13407 standard, UCD begins with a thorough understanding of the needs and requirements of the users. The benefits can include increased productivity, enhanced quality of work, reductions in support and training costs, and improved user satisfaction (Maguire and Bevan 2002). Requirements analysis is not a simple process. There are many methodologies for user requirement analysis that can be used to achieve these goals: stakeholder analysis, context of use analysis, video recording, focus groups, interviewing, scenarios and use cases, storyboards, etc. (Maguire and Bevan 2002).

Storyboards are a key and efficient means to communicate results of user needs analysis to the team members of a multidisciplinary group of professionals involved in user-centred software engineering (UCSE) projects (Haesen et al. 2009). Storyboards contain sketched information of users, activities, devices and the context of a future application.

An additional challenge when collaborating within a multidisciplinary UCD team is communication within the team without information loss. One missing link in most user-centred processes is an approach and accompanying tool to progress from informal design artefacts (e.g. scenarios) towards more structured and formal design strategies (e.g. task models, abstract user interface designs) without losing any information. Existing tools and techniques often require specific knowledge about specialised notations or models, and thus exclude team members not familiar with these notations or models. Furthermore, functional information may be missing in informal design products, while structured design results may not always contain all non-functional information. Storyboards are a comprehensible notation that allows these shortcomings to be overcome (Haesen et al. 2016). Storyboarding has also

proved to be a good methodology for developing interactive multimedia applications such as video games (Lambert and Jacobsen 2015).

A storyboard graphically represents a sequence of actions or events that the user and the system being designed go through to achieve a task. A scenario is one textual story about how a product may be used to achieve the task. It is therefore possible to generate a storyboard from a scenario by breaking the scenario into a series of steps which focus on interaction and creating one scene in the storyboard for each step (Rogers et al. 2011). The purpose for doing this is twofold:

1. To produce a storyboard that can be used to get feedback from users and colleagues.
2. To prompt the design team to consider the scenario and the product's use in more detail.

After this process is finished, the technical team can process the stories and develop a draft that is focused on the way the user will interact with a hypothetical application. This application implements the interactions portrayed in the storyboards. After the technical team validates this draft, the rest of stakeholders can evaluate it. Stakeholders can provide feedback with new suggestions, objections or confirmations. The technical team process this feedback and produce a prototype of the application that will have all the main features of the definitive application, taking into account the suggestions of the stakeholders. A number of cycles of design-test-implement are performed in order to refine the visual aspect of the application, workflows, ergonomics, etc. before developing the final system.

2 Using Design Thinking in AiRT

The main goal of the AiRT project is to provide the European creative industries with a new tool that will enable them to offer new services and to grow in the international market. To achieve this objective, we have designed the first RPAS that is specially designed for professional indoor use (Santamarina-Campos et al. 2018).

Inclusive and participatory methodologies and work tools have been used in order to achieve the specific results and objectives of the project. These methodologies have allowed collaboration and communication both internally (between the consortium) and externally (with the end-users) of the project. For this reason, Design Thinking methodology (Both 2009) has been used to ensure that:

1. Results are aligned with the values of the creative industries (García and Dacko 2015) from the initial stages of innovation processes.
2. Communication between the interdisciplinary team that makes up the consortium (composed of engineers, economists, artists, creatives, etc.) is effective.

This methodology (see Fig. 1) is centred on the user experience, focusing on the design process rather than the final product. It allows the convergence of different fields combined through “radical collaboration”, with the common goal of



Fig. 1 Methodologies used in the design of the Ground Control System (GCS) software. Source: Own elaboration, adapted from Both (2009)

implementing tools that enable new flows of thought based on intuition, critical thinking and creativity (Brown 2009).

Thereby, storyboarding can be found in the intersection between narrative, visual thinking, and design thinking (Beckman and Barry 2007; Wikström and Berglund 2011).

3 Methodology Development at AiRT

Design Thinking (Both 2009) has allowed the consortium to be connected to the experience of the creative industries. Creative Industries professionals have participated in the identification of needs and have interacted with the prototype during demonstrations. The aim was to learn from user feedback. Design Thinking is a

suitable methodology for this interdisciplinary consortium, formed by engineers, managers and creatives, and from the integration of experts from 13 sectors of the European creative industry (Santamarina-Campos et al. 2018). The five steps are analysed and developed below.

1st Phase. Empathise

The project began with the analysis of the needs of the creative industries. This analysis was carried out with end-users, which allowed us to obtain consistent results. The technique of the Stakeholder Map was used to identify the potential users of the product. It allowed us to have a clear image of the users who had to intervene and participate in the analysis. Based on the stakeholder map defined by the consortium, the key informants who participated in the definition of the features of the system were identified through three focus groups in Spain, Belgium and the United Kingdom. The dynamics were prepared by means of a previous Documentary Investigation around the requirements related to aerial filming and photography, the use of drones and the security and data protection problems derived from them. The key informants chosen at this stage came from 13 different sectors of the creative industries and they participated in the entire project process.

2nd Phase. Synthesise

An analysis of the needs of the CIs and ethical and risk issues was carried out based on the information extracted in the previous phase. Qualitative Content Analysis, Social Network Analysis (SNA), and manual coding and categorisation of qualitative data were the methods used for processing the gathered information. The first step was to identify the real needs of the end-users. These needs drove us to define possible key solutions that provided added value and let us obtain an innovative result. The results obtained at this stage were the basis of the idea of the AiRT system.

3rd Phase. Ideation

The target of the ideation stage is that both CIs and creative professionals, who are part of the consortium, can define the functionalities to be implemented in the AiRT system. The AiRT system is composed of an RPAS that is driven automatically from land by a Ground Control System (GCS) software. The GCS runs on a standard tablet. Written scripts (Fig. 2) were elaborated, starting from the identification of the needs carried out in the initial phase, together with the specifications included in Annex 1 (part A) of the DoA,¹ in the Grant Agreement no. 732433.

These written scripts were subsequently transferred to graphic scripts (Storyboards) (Fig. 3) that represented the use of the AiRT system in different creative scenarios. Storyboarding was one of the main techniques used during the process of requirements elicitation. It allowed great visual and plastic content. Storyboarding eases creative and analytical thinking while facilitating communication between internal groups of the consortium. Innovative and feasible solutions arise around

¹Description of the Action.



WPS Task 5.1 Software adaption - user interface programming
D5.1: End-user friendly adapted software

STORYBOARD – AiRT SYSTEM

SYNOPSIS

A young filmmaker is filming his first feature film. One of the main scenes of his first work will be filmed inside an old textile factory of the early twentieth century. The chosen space as a stage, is a large abandoned industrial building of 1000 m² of rectangular porticoed plant. It presents two rows of fine columns, typical of textile factories of that time, that run along the length of the room dividing it into three equal sections. The deck is flat, with exposed beams, with a height of 4.5 meters. It features small sized spans located at the top of one of the sides of the ship, allowing controlled entry of natural light, creating an intimate atmosphere. It features small sized spans located at the top of one of the sides of the industrial building, which allows controlled entry of natural light, creating an intimate atmosphere. The scene to be shot represents the game of two five-year-old children between the columns. This scene will not only represent the film, but also be used for the design of the poster, thus it will be necessary to take photographs. It tries to obtain a free movement of the camera, which accompanies the freedom of the children's movement. Therefore, a not invasive tool is needed, so that the children can move with freedom and spontaneity, that allows to obtain continuous and clean shots of the space. That is why the use of rails, cranes, and other auxiliary means that may obstruct the space and limit the free movement of the camera is discarded. On the other hand, the recording should be done in the blue hour, so the shooting time will be very brief and will not allow the repetition of shots.



Technology transfer of Remotely Piloted Aircraft Systems (RPAS) for the creative industry - AiRT
www.airt.eu



WPS Task 5.1 Software adaption - user interface programming
D5.1: End-user friendly adapted software

VISUAL PLANIFICATION

Time Line	Type of shot	Scene/ Utility/ Actors	Shot description	Dialogues
1	Long shot zenith in the back of the characters	Interior textile factory of 1900. Natural lighting through the windows, 13:00 hours. Director Cameraman Pilot Assist Drone AiRT	In the abandoned textile factory, the director, the Cameraman and the Pilot Assist meet the day before the shooting of the scene. They are in the centre of the hall, around the drone AiRT. Each person has some physical trait that characterizes him. On their shirts, the role they play in the film is indicated.	Text at the bottom of the bullet: Day before the filming of the main scene of the film, the director, the Cameraman and the Pilot Assist meet at the stage chosen to prepare the filming. Balloons: Cameraman: The location is perfect: a labyrinth of columns that will accompany the children's game. Director: The shot with the children can not be repeated, it has to be perfect at first try, to collect the spontaneity and freedom of movement of the children. Remember that we will film in the blue hour. Cameraman: So, it is fundamental that we have very clear the path of the camera.
2	Long shot zenith head on the characters	Director Cameraman Pilot Assist Drone AiRT	In the long shot, slightly zenith of the industrial building, the characters are seen from the front and the anchors are highlighted. In a corner, a close-up of a tablet is shown with a finger marking "calibrate". The Pilot Assist presses the calibration button on the Tablet and the drone moves in the "Z" axis half a meter up, and makes a 360 ° turn without barely moving from the take-off place.	Text at the bottom of the bullet: AiRT is ready, the application shows at the top of the window a timeline, which indicates sequentially the necessary steps to reach the final flight. At the moment the timeline runs to the first milestone indicating that the equipment is ready, waiting to comply with the next step that will be the calibration. Balloons: Pilot Assist: ... So, first thing, let's prepare the equipment, I've already put 4 anchors for the IPS. It calibrates in a second. Look ... can you see it? This green flashing button indicates that everything is ok and we can do the calibration, press here and ... ready! And now we can start with the mapping. Director: Perfect! How long will it take you to do the mapping? Pilot Assist: It will be fast, because it does not have much height. In three passes it will be ready, one pass for a meter and a half. Cameraman: Does the mapping have only one flight option? Pilot Assist: No, it has three, depending on the support you need. Look, take the Tablet and you'll see how I do it.

Technology transfer of Remotely Piloted Aircraft Systems (RPAS) for the creative industry - AiRT
www.airt.eu

Fig. 2 Examples of written scripts that describe a hypothetical storyboard scenario. Source: Own elaboration

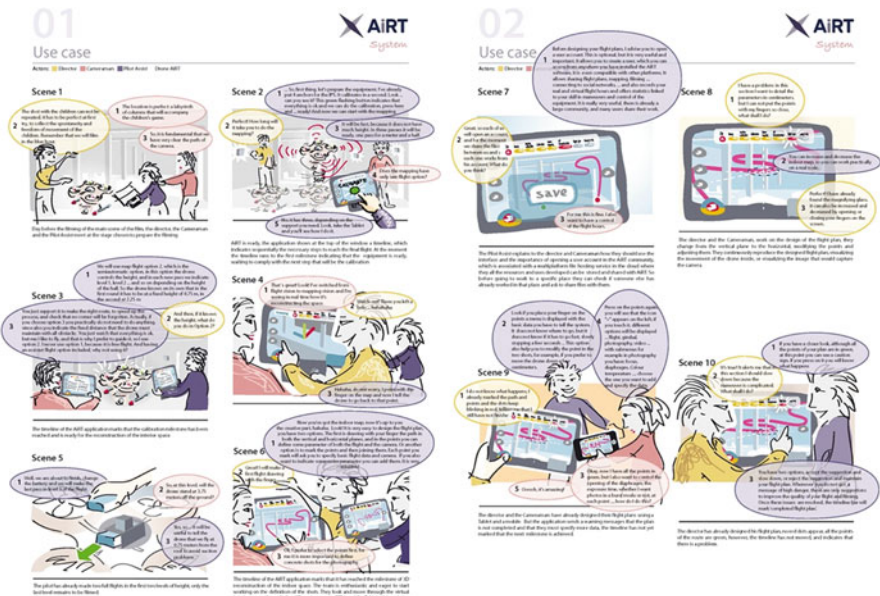


Fig. 3 Collection of the graphic script covering a storyboard to define possible innovative solutions after identifying real needs of the end-users. Source: own elaboration

storyboards. The use of this tool has allowed us to fulfil one of the main premises of the creative process of Design Thinking, “Show, don’t tell” (Plattner 2010). This means that it is important to communicate the vision in an impacting and meaningful way by creating experiences, using illustrative visuals and telling good stories.

The use of this tool favoured expansive thinking. Programmers could communicate the main ideas that were drawn from the previous phase directly and more clearly, without value prejudices. The purpose of the use of storyboards was to help the creatives involved in the consortium to transfer real scenarios to the design process of the AiRT system. These scenarios created possible indoor spaces that would reflect the identified needs in different creative spaces.

An heuristic analysis of the main programmes for flight plans based on mesh or mosaic was performed in this phase. This analysis was also done for the best mapping and photogrammetry programmes available on the market. The goal of this study was to analyse existing solutions similar to our product in order to have a more complete perspective of issues around the usability and final design of our tool.

To conclude this phase, a storytelling of the history of the project was elaborated, with the target of informing the creative industries and other sectors of the potential of the tool.

4th Phase. Prototype

In this phase, the ideas expressed in the previous stage were materialised. Bear in mind that the development of prototypes is not simply a way to validate ideas, but it

Fig. 4 Participation, action and research interoperation generates PAR methodology. Source: own elaboration



is an integral part of the innovation process (Plattner 2010). Therefore, this phase carries out many other implicit goals. The AiRT RPAS is a complex system that involves a high-end Indoor Positioning System (IPS) technology that was in beta stage of development. AiRT RPAS can be seen as a real-world test bench for the IPS. A new drone was also developed in order to match the requirements of an indoor drone with support for professional cameras. This drone had its own standard Flight Control System (FCS). Finally, the Ground Control System (GCS) connects wirelessly to an On-board Control System (OCS). During this phase, all components of the system, FCS, OCS, GCS and IPS were integrated. Meanwhile, a software prototype that implemented the functionalities of the AiRT system was developed. This software runs on the GCS. It has to communicate bidirectionally with the server process running on the OCS. The design of this software was based on prioritised requirements, with the goal of making the solutions visible. The Human-Computer Interface of the software (user interface) was based on the heuristic analysis of the information extracted from the storyboards done in the previous stage.

Potential improvements were identified, but they were relegated for later developments, after users had a chance to interact with the product and provide feedback.

5th Phase. Tests

During this stage, end-users in the three participating countries tested prototypes from a selection of scenarios relevant to the creative industries. The objective of this stage was to involve users in the identification of failures or in the contribution of new improvements, through the **Participatory Action Research** tool (PAR) (Chevalier and Buckles 2013). PAR focuses on the effort to integrate three main aspects: Participation of all the stakeholders, Action as an engagement with experience and history and Research to extract the knowledge (Fig. 4). The purpose of this type of technique is to obtain relevant data from key informants that allow subsequent interpretation and analysis of the facts from the experiences (Santamarina-Campos

et al. 2017). In parallel, the dynamics were recorded with the objective of using the “covert observation technique” (Bryman 2016), through the analysis of the filming of the sessions, and then interpretation was done using qualitative data analysis software.

A workshop is also another activity for making the tool known to the CIs among other sectors. The aim is to bring AiRT closer to the market. Finally, two storytelling lines will be developed:

1. One will show the whole process of the project, from the ideation phase to test.
2. Another one for commercial purposes only.

4 Requirements Elicitation and User Interface Design in AiRT

Requirements analysis is a set of activities performed to determine the needs or features that a new product should have. It is an important stage in any software engineering project. Requirements analysis is critical to the success of a software project. It helps to identify business needs or opportunities. It defines the functionality of a programme to a level of detail that is high enough to perform a system design. The requirements should be documented. Every requirement has to be:

1. *Feasible*: it can be accomplished by a computer program.
2. *Measurable*: it is computable and can be translated into numbers in some way.
3. *Validatable*: the development team has to test and check that what the programme does is what the requirements states.
4. *Traceable*: there is always a user need that justifies any action that the programme does. On the other side, the programme does not do anything that has not been requested by the customers.

Requirements analysis includes three types of activities that have been used for this project:

1. *Eliciting*. Requirements gathering or requirements discovery are performed by analysing documentation of the business process and by interviews with the stakeholders. After these meetings, the technical team generates an intermediate documentation consisting of some user stories and storyboards.
2. *Analysis*. A phase where the team determines whether the informal requirements obtained in the previous stages are clear, complete, consistent and unambiguous. Sometimes these requirements cannot meet everyone’s expectations. So, some decisions have to be taken in order to resolve any apparent conflicts.
3. *Recording*. This phase is in charge of documenting the requirements including a summary, natural-language documents, storyboards and anything else that can affect any other area of the software production: data models, communication models, process specifications, etc. The software requirements specification

document is the distillation product that emanates from all the requirements analysis process.

The first activity, eliciting was performed by using focus groups as seen in the 1st phase, Empathise. The second activity, Analysis, was made by using storyboards, as seen in the third phase of Ideation. The third activity, Recording, tried to detect in a first step the main areas where the requirements could be assigned. We used collaborative technologies for the development team. This allowed us to perform asynchronous collaboration, effective comments exchange, the integration and unification of definitions and the reduction of redundancy.

Although the goal of this phase was to obtain a list of user interface (UI) requirements, during the analysis many other requirements not related directly to UI appeared. These involved programming and communications requirements that emanated from the analysis of the user interface and software functionality analysis. The latter types of requirements are especially important for this project since they involve all the three development teams: software, IPS and drone development. These requirements are for internal use only and they were reserved for later use by the development team.

The storyboard scenarios, as described above, are descriptions of practical situations where professionals use the drone for recording needs. Most user requirements are obtained from storyboards. Several storyboards were proposed and catalogued.

Requirements have to be traceable. This means that every requirement has to be connected to the storyboard where it emerged, allowing for the quick location of both requirements and storyboards. We provided a unique identifier for each requirement and for each storyboard. References to storyboards follow the syntax:

$$SourceType + XX + ":" + page$$

where

SourceType is the type of document where the storyboard is described. For instance, UC as an acronym for User Case, but there are other types of documents used during the analysis stage.

XX is the number of the storyboard.

page is the page number where the requirement appears inside of the storyboard.

After analysing the storyboards, the requirements are compiled on a list. This list may change during the development phase due to user feedback or the discovery of new emerging functionality that can complement or clarify some ambiguous requirements. After some time, and with the whole consortium's agreement, the requirements list is frozen. This list defines the initial functionality of the system. The goal of this step is avoiding requirements creep, that is, uncontrolled growth in the project's scope during the development phase.

As the requirements were identified, they were also classified. Related requirements were grouped into categories. Table 1 shows the categories used in the project.

Table 1 Definition of categories

Category name	Acronym	Description
Safety	SAFE	Security and safety. The drone has to fly and return home without incident or endangering its integrity or people in the flight area or nearby.
Power	POWE	Features regarding energy spent by the system and power up/shutdown procedures.
Looks and design	LOOK	Appearance of the drone. Hardware design: housing, careen, etc.
Pre-production	PRE	Anything happening before the CALI step.
Calibration	CALI	A phase previous to RECO where the IPS bases are set and recognised by the system in order to allow a precise localisation of the drone.
Recognition	RECO	Flight phase where the flight environment is scanned in order to provide an environment for helping during the design of the recording flight plan.
Flight plan	FPLA	Process of creating the flight plan in the virtual environment map: definition of waypoints, path edition, etc.
Recording	REC	Execution of the selected flight plan over the real space for recording the video or taking photos.
Post production	POST	Anything happening after the REC step.
In flight	FLY	Applies whenever the drone is flying (for any purpose).

Source: own elaboration

Each category represents a state of the application workflow, or an area of the system.

A requirement can be assigned to:

- A single category. For instance, a requirement such as “the initial calibration of the positioning system must be performed automatically” has been assigned to the category CALI.
- Several categories. For instance, a requirement such as “the drone must not collide with any wall when performing the environment 3D mapping” may be assigned both to categories RECO and SAFE. Whenever a requirement is classified in more than one category, one of them is chosen as the primary.

The description of each requirement is organised in several fields (see Fig. 5):

- *Identifier* (ID). It provides an identity (name) to the requirement. It follows the convention of RXXX where R is the short for “requirement” and XXX is the number assigned to this current requirement. Requirement IDs are never reused. Therefore, when two different requirements are found to overlap, one of the two IDs is deleted and never used again, and its requirement is assigned to the remaining IDs. Therefore, the final list of requirements will have gaps in the requirements IDs. This process helps with the requirement traceability.

ID	R004
Category	FPLA
Description	The application has to know or ask for the predefined operational limits of the drone (maximum speed, maximum acceleration, etc.), the gimbal and the RCAM The application has a file or a configuration which can be used to load these values. The same file or configuration should be loaded into the server on the OCS.
Actors	Client application, OCS
Source Storyboards	UC03:1, UC04:1, UC06:1, UC08:1, UC08:2, UC09:2, D33:7, D33:17
Precondition	This process starts when the application starts.
Comments	All of the drone main telemetry parameters are sent from the FCS to the OCS. These data are sent back to the GCS in order to be shown at the UI if required.
Responsible	AEROTOOLS
Version	1.0

ID	R026
Category	SAFE
Description	The user can specify the minimum safety distance to obstacles. These values should be taken into account to perform the flight at “Mapping” step or at “Recording” step.
Actors	Client application, OCS, FCS
Source Storyboards	UC00:9
Precondition	Define settings values is the first step before the drone takes off.
Comments	The values introduced by the user have to be in a valid interval. The safety distance to obstacles (walls) is needed to calculate the different heights at which the mapping could be done. These requirements have to be sent to the OCS in order to warn FCS to change trajectory in case security is compromised.
Responsible	UPV
Version	1.0

Fig. 5 Examples of requirement definition. Source: own elaboration

- *Category*. The category assigned to the requirement. It follows the code provided above.
- *Description*. An explanation of the goal of the requirement. This list contains high-level requirements that, during development, may be subdivided into requirements that are more detailed.
- *Storyboards*. This field enables us to relate each requirement with the storyboard it derived from, and therefore to trace it. There must be at least one customer need that justifies any function implemented in the system. There are many internal technical requirements that were not explicitly mentioned by the customers

because of the customer's lack of technical knowledge. These requirements have to also be traceable and have to appear in the list, although they are secondary requirements made by the development team. In any case, there is always a reference to the code assigned to the storyboard that justifies the current requirement.

- *Responsible*. It is the team or the person in charge of developing this requirement. They have to supervise or execute the correct implementation of the requirement.
- *Comments*. They allow expanding or refining a bit more the description of the requirement. This field is not mandatory.
- *Version*. It indicates the assigned version of the software when the requirement will be implemented. Current version of the software is V1.0. If the requirement is assigned to the first version of the drone, the requirement is mandatory. It has to be developed within the project scope. If the requirement is not important or it is not clear if industry will demand it urgently, it is set to a later version. If there is time enough to complete it, perhaps it will be developed and integrated in the current version. If the requirement demands a lot of work and it is not clear that the development could complete the requirement in time, it will be assigned to V2.0 for the future development of the drone.

5 Conclusions

Requirements Analysis is a complex process that involves users and developers during most of the software development cycle. At the beginning of the project, requirements are obtained using different techniques with all the stakeholders. Later, during development, requirements are refined and adapted to the development evolution, and it is necessary to keep users updated and take their feedback into account.

We obtained and analysed the real needs of our users through three focus groups. We found that current indoor drones do not provide the features that creative industries demand, and thus we were able to detect many possibilities for the AiRT system. At this point, the use of storyboards made it possible to simplify and add a narrative and an appropriate context that eased the design process.

Storyboards provided visuals for the consortium's developers with real uses and potential user experiences, based on the identification of needs. Storyboards were an invaluable tool for developing the requirement list that describes the specific functionalities to be implemented in the AiRT system, and specifically in the GCS software. The graphic scripts helped to promote communication between software developers and the consortium's creatives.

Many GUI requirements were specific for this system since there is no comparable product on the market. CI users expressed their requirements from previous experiences with drones in outdoor scenarios. They also expressed:

- their problems in moving auxiliary equipment for filming indoors, especially in inhabited spaces or places with heritage value (Informant 6 and 8);
- invasive aspects involved in the use of auxiliary means such as lifting platforms, travel or sliders (Informant 1);
- the time-dependent condition of filming or photographing indoors with natural light and the difficulty of repeating identical shots (Informant 19);
- the limitations of auxiliary means of obtaining special footage (Informant 20); and
- experiences of using indoor drones, highlighting the need to incorporate an indoor positioning system and safety measures (Informant 13).

On the other hand, although current technology for outdoor drones is mature, this technology does not match the more demanding indoor requirements. Therefore, we had to push this technology to its end in order to reach the minimum specifications considered essential for the CI.

After the analysis of the user requirements, there were some additional technical requirements that arose because of the demands of future users. These requirements were mainly about:

- how the device is introduced in the industry's workflow;
- security specifications, some mandatory, some complementary;
- error control when flying; and
- other difficulties that appeared when implementing in restricted hardware platforms, for instance, the amount of geometry information that can be stored by the OCS depending on the memory available, transfer speed depending on the Wi-Fi network, rendering limits of the tablets used for the project, etc.

Putting the specifications together, we found that some were not feasible in the current state of drone technology. Others were also not feasible because they go beyond the project objectives and thus cannot be implemented with the given timeline and resources. So, some requirements have been postponed for later versions.

These specifications may still change, especially as beta versions of the system start to be used by the final users. Then they will realise the potential of the indoor drones and start to suggest more changes concerning the GUI.

References

- Both T (2009) Bootcamp Bootleg. <https://dschool.stanford.edu/resources/the-bootcamp-bootleg>
- Brown T (2009) Change by design: how design thinking can transform organizations and inspire innovation. HarperCollins, New York
- Beckman SL, Barry M (2007) Innovation as a learning process: embedding design thinking. *Calif Manage Rev* 50(1):25–56
- Bryman A (2016) Social research methods. Oxford University Press, Oxford
- Chevalier JM, Buckles DJ (2013) Participatory action research: theory and methods for engaged inquiry. Routledge, Abingdon

- Finch C (2011) *The art of Walt Disney: from Mickey Mouse to the Magic Kingdoms and beyond*. Harry N. Abrams, New York
- García R, Dacko S (2015) Design thinking for sustainability. In: Swan S, Griffin A, Luchs MG (eds) *Design thinking: new product development essentials from the PDMA*. Wiley, New Jersey, pp 381–400
- Haesen M, Vanacken D, Luyten K, Coninx K (2009) Supporting multidisciplinary teams and early design stages using storyboards. In: Jacko JA (ed) *Human-computer interaction. New Trends, HCI 2009*. Springer, Berlin, pp 616–623
- Haesen M, Vanacken D, Luyten K, Coninx K (2016) Storyboards as a Lingua Franca in multidisciplinary design teams. In: *Collaboration in creative design*. Springer, Cham, pp 211–231
- ISO/IEE 24765-2010(E) (2010) *Systems and software engineering, vocabulary*. <https://www.iso.org/standard/50518.html>. Accessed 16 Mar 2018
- Lambert D, Jacobsen M (2015) Building digital video games at school: a design-based study. In: Preciado BP, Takeuchi M, Lock J (eds) *Proceedings of the IDEAS: designing responsive pedagogy conference*. Werklund School of Education, University of Calgary, pp 32–42
- Maguire M, Bevan N (2002) User requirements analysis. A review of supporting methods. In: Baeza-Yates RA, Montanari U, Santoro N (eds) *Proceedings of IFIP 17th world computer congress*. Kluwer Academic, Montreal, pp 133–148
- Plattner H (2010) *Stanford D. School Bootcamp Bootleg*. Available at Institute of Design Thinking at Stanford. <http://dschool.stanford.edu/wp-content/uploads/2011/03/BootcampBootleg2010v2SLIM.pdf>. Accessed 15 Feb 2018
- Rogers Y, Preece J, Sharp H (2011) *Interaction design: beyond human-computer interaction*. Wiley, New York
- Santamarina-Campos V, De-Miguel-Molina B, Segarra-Oña M, De-Miguel-Molina M (2018) Importance of indoor aerial filming for creative industries (CIs): looking towards the future. In: Katsoni V, Velander K (eds) *Innovative approaches to tourism and leisure*. Springer, Cham, pp 51–66
- Santamarina-Campos V, Martínez-Carazo EM, Carabal-Montagud MA, De-Miguel-Molina M (2017) Participatory action research (PAR) in contemporary community art. In: Santamarina-Campos V, Carabal-Montagud MA, De-Miguel-Molina M, De-Miguel-Molina B (eds) *Conservation, tourism, and identity of contemporary community art*. Apple Academic Press Taylor & Francis, New Jersey, pp 105–136
- Taylor A (2013) *Design essentials for the motion media artist: a practical guide to principles and techniques*. Taylor & Francis, New York
- Wikström A, Berglund K (2011) A narrative approach towards understanding innovation. In: *Proceedings of 18th international product development management conference*, 2011

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

