International Journal for Innovation Education and Research

ONLINE ISSN: 2411-2933 PRINT - ISSN: 2411-3123

A Collaborative System Architecture for Developing Digital Electronic Games

Filipe de Carvalho Pinto Raulino;Raiane dos Santos Martins;Akynara Aglaé Rodrigues Santos da Silva Burlamaqui;Aquiles Medeiros Filgueira Burlamaqui

Abstract

Educational electronic games have been gaining lots of space inside classrooms because of their contribution to the learning and motivation of the students. However, the development process of a game is complex and, normally, involves many people with different profiles, making it impossible for the teachers to develop games according to their pedagogical needs. Even though the existence of technical barriers, the teacher's involvement in this process is fundamental, since they are the ones that have the expertise on the pedagogical components of the process. Through a collaborative process, it is possible to merge all needed elements for the development of educational electronic games, since it allows mutual help among the participants. Furthermore, providing individual learning to each collaborator, through the exchange of ideas and experience gain throughout the project development. This article has the objective of creating a community for collaborative development, that includes the pedagogical features on the development of educational electronic games. For this we will present a collaborative system model, based on the roots of the 3C collaboration model: communication, coordination and cooperation.

Keyword: Collaborative systems; educational games;

Published Date: 2/28/2019 **Page.154-166** **Vol 7 No 2 2019**

DOI: https://doi.org/10.31686/ijier.Vol7.Iss2.1334

A Collaborative System Architecture for Developing Digital Electronic Games

Filipe de Carvalho Pinto Raulino (Corresponding author)
Universidade Federal do Rio Grande do Norte
Natal (RN), Brazil

Raiane dos Santos Martins
Universidade Federal do Rio Grande do Norte
Natal (RN), Brazil

Akynara Aglaé Rodrigues Santos da Silva Burlamaqui
Universidade Federal Rural do Semi-Árido
Angicos (RN), Brazil

Aquiles Medeiros Filgueira Burlamaqui
Universidade Federal do Rio Grande do Norte
Natal (RN), Brazil

Abstract

Educational electronic games have been gaining lots of space inside classrooms because of their contribution to the learning and motivation of the students. However, the development process of a game is complex and, normally, involves many people with different profiles, making it impossible for the teachers to develop games according to their pedagogical needs. Even though the existence of technical barriers, the teacher's involvement in this process is fundamental, since they are the ones that have the expertise on the pedagogical components of the process. Through a collaborative process, it is possible to merge all needed elements for the development of educational electronic games, since it allows mutual help among the participants. Furthermore providing individual learning to each collaborator, through the exchange of ideas and experience gain throughout the project development. This article has the objective of creating a community for collaborative development, that includes the pedagogical features on the development of educational electronic games. For this we will present a collaborative system model, based on the roots of the 3C collaboration model: communication, coordination and cooperation.

Keywords: Collaborative systems; educational games;

1. Introduction

The development of information and communications technologies (ICT) in the last decades has changed the lifestyle of society by a large amount, since we have started to act by the simplicity inherent in technology. Currently we have an environment for exchanging ideas, knowledge and opinions at an unprecedented speed rate.

The sociocultural relations and the professional profiles demanded are directly affected by the merge of technologies to the common routine, demanding an update on the curricula and the traditional teaching and learning manners. Almeida and Valente (2011) said that the ICT must be used to further potencialize the pedagogical practice allowing the development of the student's autonomy on the search of meaningful information and on the development of critical thinking.

Education must be in sync with the others aspects of the student's life so that their curiosity can be pushed forward. Therefore, the ICTs have the possibility of promoting an environment favorable to learning. Tarouco et al. (2004) point out educational games as a complementary tool on the construction and retention of concepts developed on classroom, as well as a motivator both for the student and the professor.

However, the production of educational games needs, besides pedagogical planning, knowledge of multiple technologies, such as high-level programming languages, what requires a multidisciplinary team and, consequently, a high production cost, many times making the educator close the production by themselves because of their pedagogical needs.

This scenario ends up forcing the educator to not use or to adapt their content and didactic to the available material, instead of customizing in agreement to their necessity or the one of a group of students.

Against the exposed scenario, one can realize that there is a demand for the insertion of new educational technologies in the classroom, and that there is an obstacle made by the lack of technical knowledge and on the complexity of the development of these artifacts, especially with regard to educational games.

Still, many educators have trouble in using these resources on the classroom, especially when it comes to digital games, because they aren't able to find games suitable to their needs. Almeida and Valente (2011) have stated that one of the key factors that contributed to the separation of ICTs from curricular activities was the fact that the used softwares were far from what is discussed on classrooms.

To create a game is not a trivial task, besides creativity and a good script, it is needed to create, edit and organize many types of media and the development of the programming code, in order to create the game's dynamic and control the interaction with the user. The characteristic complexity of the gaming production implies on the necessity of a series of professional profiles such as programmers, illustrators, animators, musicians, game designers, among others according to the nature of each game. In the event of educational games there are still educators and pedagogues. Bertin (2015) et al. stated that the development of *gamified* educational technologies constitutes an interdisciplinary challenge that involves in-depth studies in Didatics, in Games and in Software Engineering.

To adjust the alignment between the pedagogical and techniques is one of the main faced challenges. To Tarouco et al. (2004) the ideal solution would be that the teachers could create their own games, adapting them to their contexts without compromising pedagogical quality.

Therefore it is needed to work around the technical obstacles and make possible the teacher's

participation in the developing process, with the goal to prioritize pedagogical components. This article proposes the creation of a collaborative system for the production of educational electronic games that makes possible the connection between educators and the tech team (developers, designers), in order to contribute for the creation of didactically relevant games and to make available an open digital games repository.

2. 3 collaboration model

The proposed idea that initiated the 3C model was formulated by Ellis et al. (1991) and classified the implementation of a collaborative activity in three domains: communication, coordination and collaboration. Later, other authors (Amiour & Estublier, 1998; Borghoff and Schlichter, 2000; Fuks, Gerosa and Lucena, 2002) started to adopt the term cooperation in place of collaboration that went on to designate the whole process of working together. (Fuks et al., 2011).

The model is based on the premise that communication refers to information sharing and negotiation of commitments between subjects, coordination involves the part of managing people, activities and resources with the purpose of resolving conflicts and organize to avoid the waste of effort, and cooperation is related to the involvement of group members in achieving the common goal (Correia, 2011).

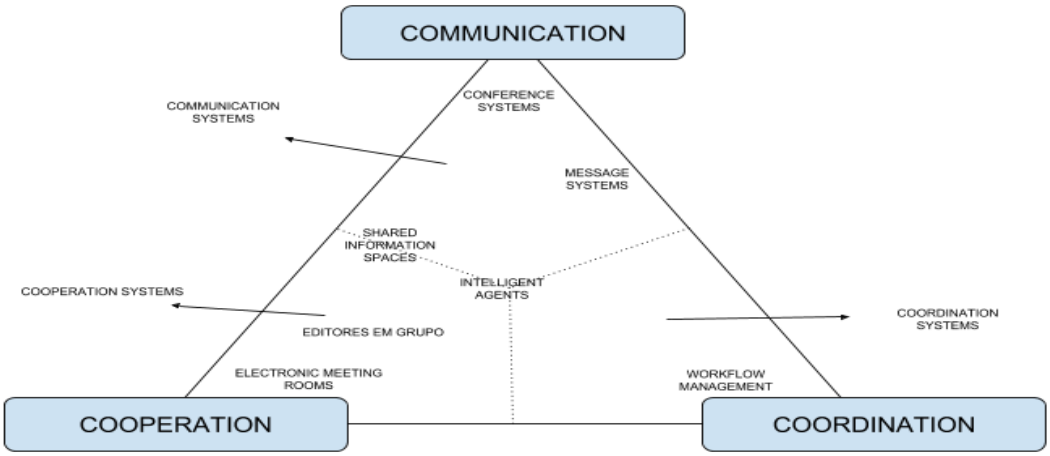

Figure 1. Classification of systems according to 3C model (adapted from Borghoff and Schlichter, 2000)

As we can see in Figure 1, collaborative systems are positioned in a triangular space where vertices are the collaborative domains, the positioning of collaborative systems in the triangular space emphasizes the contiguity between the 3Cs. Although the objective of a tool can be turned to the specific support of one of the Cs, it will still contemplate aspects of the other Cs (PIMENTEL et al., 2006).

2.1 Communication

Communication involves negotiation and establishing compromise. A computer-mediated communication tool supports interactions among participants, and can manage state transitions, dialogue events, and the commitments of each participant (Fuks, Raposo and Gerosa, 2003).

The communication represents an action between the sender whom, according to their goals and commitments, produces and sends a message to the recipient that, when receiving and interpreting the message, can cause their commitments and knowledge to be modified. In order to transmit the content of the information, the sender transmits signals in an appropriate and perceptible language for interaction with the receiver, so that everyone can perceive it (Costa, Loureiro and Reis, 2014).

There are several purposes for group communications, Fuks, Raposo and Gerosa (2003) affirm that the tools of asynchronous communication are used when one wishes to value the reflection of the receivers, because they will have more time to act. Synchronous tools value the speed of interaction by providing immediate communication. Communication will succeed if the sender's goal results in the expected commitments. The only way to obtain evidence of successful communication is through the receiver's speech and actions.

2.2 Coordination

Coordination is responsible for organizing the tasks to meet the commitments generated by the communication, assigning tasks, objectives and constraints so that efforts are not wasted on repeated or conflicting tasks.

Coordination aims at ensuring the productivity and success of the group's objectives, keeping the group cohesive, the participants aware of their role in the group and focused on their specific objectives and the group's overall goal (Santos, 2006). Coordination is essential to keep the group together and motivated to achieve common goals.

Fuks, Raposo and Gerosa (2002) emphasize that perceptual information is essential for group coordination. It is important that each one knows the progress of the work of their companions: what was done, how it was done, what is missing for the conclusion, what are the preliminary results, and so on.

2.3 Cooperation

Tasks originate from the commitments negotiated during the communication, are managed by the coordination and are carried out during the cooperation. Through mechanisms of awareness the individual gets feedback from his actions and feedthrough the actions of his colleagues. When cooperating, it is necessary to renegotiate and make decisions on unexpected situations, which requires new rounds of communication and coordination (Gerosa, 2006).

Cooperation is directly associated with the joint execution of tasks by group members. For this, an environment is necessary that allows to generate and manipulate shared artifacts, essential for the accomplishment of the objectives of the tasks (Santos, 2006). Cooperative environments should provide appropriate tools for the accomplishment of the tasks, such as co-authoring tools, for example.

2.4 Awareness

Fuks, Raposo and Gerosa (2003) also cite awareness as an essential concept in the 3C model, which permeates the three domains of the 3C model. awareness is generated by the interactions that occur in the group, serving to mediate the entire collaboration. It is through the awareness that individuals become aware of the common goal and the role of each within the context.

awareness is an understanding of the activities of others, which provides a context for one's own activities. This context is used to ensure that individual contributions are relevant to the group's activities as a whole, and to evaluate individual actions in relation to the group's goals and progress (Dourish and Bellotti, 1992).

Perceiving the activities of other individuals is essential for the flow and naturalness of work and for reducing the feelings of impersonality and distance, common in virtual environments (Gerosa, Fuks and Lucena, 2001).

In this way, perception can be interpreted as the knowledge and understanding of everything that occurs inside and outside a group that are relevant to the development of the activities of its participants. Perception is achieved when the other people involved in an activity receive, process and understand the actions of others (Santos, 2006).

3. Collaborative System Development

Fuks, Raposo and Gerosa (2002) affirm that a collaborative system usually consists of a set of tools, which allows the interaction between multiple users. The collaborative system must be flexible to adapt to the needs of each group, enabling them to set up their work context and configure a specific set of collaborative tools that meet their needs.

Fuks, Raposo and Gerosa (2002) further claim that the use of component-based development techniques is a way to facilitate the development of groupware to make it more flexible. Considering that these techniques develop modular systems, that can be adapted and combined to meet the specific needs of each group.

3.1 Component Based Development

Component-based development addresses the development of systems by the composition of reusable components through the creation of constructive software blocks. These blocks are often called components and require patterns for interaction, composition, infrastructure and services (Hernandez, 2005).

A software component is an independent, replaceable part that communicates with the environment through interfaces. Szyperski (2002) defines a software component as a composite unit of which we know only the interfaces, specified in the form of contracts, and the context dependencies. Therefore, its implementation is encapsulated and is not directly linked to the system.

The components are defined by their interfaces, one of which defines the services provided by the component, and the second specifies what services should be available (Hernandez, 2005).

The components of collaborative systems implement tools with which individuals operate on the objects of cooperation Fuks, Raposo and Gerosa (2002). These components are coupled into an architecture following the component model, which define the communication patterns capable of incorporating and extending the component functionalities of different developers.

There are several component models such as OMG/CORBA, the Microsoft COM family and Sun’s JavaBeans/Enterprise JavaBeans (Fuks, Raposo and Gerosa, 2002; Hernandez, 2005). Recently the web

service has proved to be a model that allows several solutions to problems presented by other standards (SZPERSKI, 2002).

4. Proposed solution

This work proposes the architecture of a web system for project management of educational electronic games production. This architecture provide components aiming to offer the resources for developers and project educators to work collaboratively and on a distributed basis. This architecture group together various tools to support all areas of collaboration (communication, coordination, cooperation).

Figure 2 shows an overview of the system architecture and the relationships involved in the production of digital electronic games.

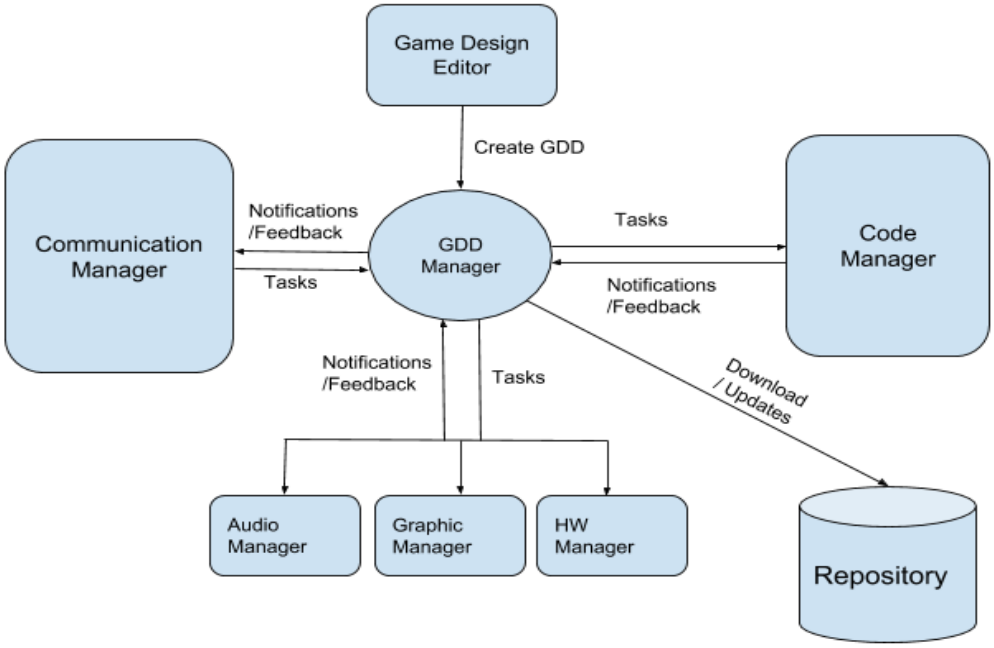

Figure 2. Overview of the system architecture

The system is completely online, available on any device that the user has access to a web browser. The environment feature an educational game description editor that will be available to anyone interested in proposing the production of a game. From it it will be possible, in a simple way, to generate a game description document, known as GDD (game design document).

The system architecture provide a collaborative environment where developers, designers, pedagogical staff and other members involved in game production can exchange information, experiences and feedback, contributing to the game's development.

From the generated GDD the developers registered in the system can select the document and start a new project or join a project already started. In the sections below you will specify some features that will be used in each of the collaboration domains. Each topic includes the components that provide the collaborative services used to support the dynamics of collaboration.

4.1 Communication services

The communication aims to provide the exchange of information, argumentation and negotiation of stakeholders. This function shall enable synchronous and asynchronous communication. The communication tools incorporated into this system are: instant messenger, forum, e-mail and comment on tasks.

The instant messenger enables synchronous communication, through text messages, with other users that are logged into the system. This tool enables group communication, with all users involved in the project, or with any specific user.

The forum maintains question and answer records divided into topics, functioning as a space for sharing experiences, exchanging ideas, solving doubts and providing a source of information.

Email provides users with the exchange of text messages asynchronously. You can attach other types of files to the message.

The comment in the tasks allows to report problems and to give suggestions during the development of the activities. It is a form of asynchronous communication for team members.

4.2 Coordination services

The coordination services have the function of managing the activities of the group. They will have activity management, notification and monitoring tools for group participation.

The activity management tool enables the creation of tasks, which can range from implementing a system requirement to solving a problem. The user must create a name for the activity and then a description and priority (high, medium or low). The task will be available to be assigned to one of the group members. Information about the activity can be updated at any time. This tool also generates the data to follow the task, such as the percentage that has already been completed, responsible and files involved.

The notification tool enables everyone in the group to receive important prompts about project progress. It passes on the information and alerts needed to organize the members' participation in the project, such as completing or adding new tasks, making new versions available or receiving messages.

The monitoring of the participation registers the progress of the activities and generates reports allowing the management of the users of the environment, also offers personal information and keeps records of the contributions of each member.

4.3 Cooperation services

The cooperation tools are responsible for the support in the execution of the tasks. These include sharing and publishing of documents, versioning tools and authoring.

Sharing of documents will provide a repository of images, audios and other documents that will be made available for the production of the games. All documents produced in the system will be part of this documentation and will be made available with Creative Commons Attribution license CC BY, which allows others to distribute, remix, adapt and create from your work, even for commercial purposes, as long as they give you due credit by the authorship.

The code will be controlled by a version control system, preferably by developers such as GIT or SVN for example. The version control system allows multiple people to collaborate on the same project simultaneously, it controls the changes avoiding the risk that they are overwritten.

4.4 System requirements

In this section we present the system requirements divided into two tables, one for functional requirements and one for non-functional requirements. the table shows the identifier of the requirement, its name, a description and the priority that can be (E) essential, (I) important or (D) desirable.

4.4.1 Functional requirements

Table 1. System functional requirements

ID	Requirement	Description	Priority
RF1	User registration	Enable the registration and maintenance of users in the system.	E
RF2	User Authentication	Users must authenticate to be able to use the system.	E
RF3	Define the user profile	Users should be ranked according to their profile in the community (educator, developer, designer). A user can have more than one profile.	D
RF4	Educational game description editor	The system should provide an interface for creating educational game design documents.	E
RF5	Manage game design files	The system should organize and make game design files available to users.	E
RF6	Creation of projects	The system should enable the creation of projects from the available game design documents.	E
RF7	Join users to projects	The system must allow the association of users with the projects in the system.	E
RF8	Creation of tasks within projects	The system should provide the creation of tasks within the projects.	E
RF9	Assigning tasks to users	The system must allow assignment of tasks to users. Activities can be assigned at the time of creation or later selected among the available	E

		activities.	
RF10	Priority classification of tasks	The system should allow the prioritization of activities. the priorities can be changed according to the evolution of the activity.	I
RF11	Monitoring project progress	The system should provide information about the progress of the project.	I
RF12	Monitoring progress of tasks	The system should provide information about the progress of the tasks.	I
RF13	Notes on tasks	The system should allow annotation of observations in tasks.	I
RF14	Providing forum for users	The system should provide for the creation of discussion forums, both public and within the project.	I
RF15	Providing chat to users	The system should provide a chat chat for users logged into the system. Chat can be started in a group or between two users.	I
RF16	Email delivery	The system must provide e-mail to all users.	I
RF17	Notification	The system must notify all users involved about relevant changes to the project or tasks.	I
RF18	Document Repository	The system must organize a repository and share all the generated documents in the production of the games (images, audios, codes ...).	E
RF19	source code control	The system must provide versioning information for the source code.	E
RF20	Availability of test versions	The system must allow the release of test versions during project development.	I
RF21	Notes on test versions.	The system should allow feedback annotations in the trial version by users.	D
RF22	Generate activity report.	The system must generate and maintain generate reports of the activities in the system, such as changes in the project, contribution of users, login, changes in files...	D
RF23	Game availability after finalization.	The system must have an area, online, to make the finished games available.	I

4.4.2 Non-functional requirements

Table 2. System Non-functional requirements

ID	Requirements	Description	Priority
NF1	Web system	The system should be web.	E
NF2	Developed in Javascript and HTML	The system should be developed with HTML and Javascript.	E
NF3	Compatible with all major browsers.	The system must support the major web browsers (Chrome, Firefox and Safari).	I
NF4	Responsive	The system should work on any device that has a compatible web browser.	I
NF5	Register log	Record all actions performed on the system.	D

4.5 System Operation

Understanding the system's operating cycle is essential to achieve the objective of this work. When finalizing the description of a game the system generates a GDD, which already includes the basic tasks for the production of the game in question. The GDD is made available and any of the collaborators can join an ongoing project or start their version of this GDD.

The project consists of a series of tasks, each task being the implementation or modification of a software functionality. At the beginning of a new project, the system will inform you the profiles of collaborators needed to develop the game in the new project, based on the basic tasks already specified in the GDD.

In addition to tasks already added automatically by GDD users can register new tasks, necessary for implementation, fixes or adaptations. Each Created, Changed, or Finished task will generate notifications for the members involved and feed the tracking data and activity report into the system.

Members can check their peers' tasks and make suggestions through notes. Whenever you have a new version the system can generate a test version. when there are no more tasks and the final version is approved by the group the game is finalized and made available to the repository.

Figure 3 shows the flowchart of the development process of a digital electronic game project in this system.

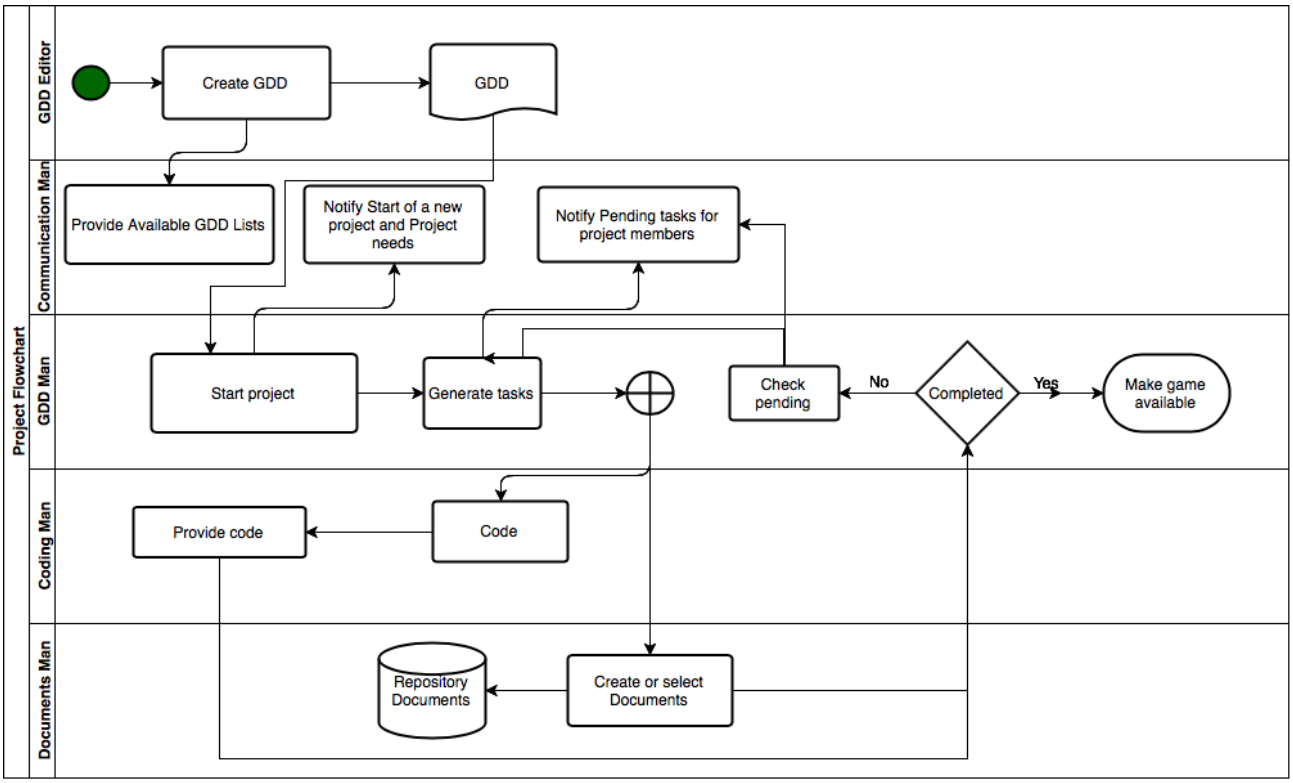

Figure 3. System flowchart

5. Conclusion

In a collaborative community individuals are encouraged to apply their personal talents in group projects. As a rule this type of community seeks, from the singular experiences, to obtain solutions to common problems. For collaboration to take place, it is necessary to offer spaces, be they physical or virtual, that will potentiate and stimulate the interaction among the members of the community.

The current context, characterized by accelerated technological development and expanded access to these technologies, favors the emergence of new paradigms, which should be used to contribute to the improvement of our society. In this sense, collaborative work can benefit greatly from the integration of ICTs, especially in the educational field. Enabling the improvement of new developers and providing access to more resources, with a strong pedagogical foundation, that favor learning and digital integration.

It is hoped that this project can provide conditions for collaborative development focused on the educational area, as well as assisting in the management of the entire software production process. Contributing to the entire community through the provision of quality open educational games and at various levels of education, offering support to educators who need technological support to put their ideas into practice and to enable learning in practice in a collaborative environment, of students interested in hone or developing skills in digital game programming and production.

6. References

[1] ALMEIDA, M. E. B. de; VALENTE, J. A. Tecnologias e currículo: trajetórias convergentes ou divergentes? São Paulo: Paulus, 2011 – (Coleção Questões Fundamentais da Educação – 10).

[2]BERTIN, Roseli; LIMA, Maria de Fátima Webber do Prado; WEBBER, Carine Geltrudes. DESENVOLVENDO JOGOS EDUCACIONAIS POR MEIO DE SOFTWARES DE AUTORIA. **Novas Tecnologias na Educação.** Cidade, p. 1-10. jun. 2015.

[3]BORGHOFF Uwe M.; SCHLICHTER, Johann H. **Computer-Supported Cooperative Work**: Introduction to Distributed Applications. Berlin: Springer-Verlag, 2000.

[4]CORREIA, António José Guilherme. **Caracterização do estado da arte de CSCW.** 2011. 187 f. Dissertação (Mestrado) - Curso de Mestrado em Tecnologias da Informação e Comunicação, Engenharias, Universidade de Trás-os-montes e Alto Douro, Vila Real, 2011.

[5]COSTA, António Pedro; LOUREIRO, Maria João e REIS, Luís Paulo. **Do Modelo 3C de Colaboração ao Modelo 4C**: Modelo de Análise de Processos de Desenvolvimento de Software Educativo. *Rev. Lusófona de Educação*. 2014, n.27, pp.181-200.

[6]DOURISH, Paul; BELLOTTI, Victoria. **Awareness and coordination in shared workspaces**. In: ACM CONFERENCE ON COMPUTER-SUPPORTED COOPERATIVE WORK, 4., 1992, Toronto, Ontario, Canada.

[7]FUKS, Hugo; GEROSA, Marco A.; LUCENA, Carlos J. P. **The Development and Application of Distance Learning on the Internet.** Open Learning - The Journal of Open and Distance Learning, v. 17, n. 1, 2002. p. 23-38.

[8]FUKS, Hugo; RAPOSO, Alberto Barbosa; GEROSA, Marco Aurélio. **Engenharia de Groupware: Desenvolvimento de Aplicações Colaborativas**, XXI Jornada de Atualização em Informática, Anais do XXII Congresso da Sociedade Brasileira de Computação, V2, Cap. 3, 2002, Rio de Janeiro, ISBN 85-88442-24-8, pp. 89-128.

[9]FUKS, Hugo; RAPOSO, Alberto Barbosa; GEROSA, Marco Aurélio. **Teorias e modelos de colaboração**.In: PIMENTEL, Mariano; FUKS, Hugo (Org.). **Sistemas colaborativos.** Rio de Janeiro: Elsevier Editora Ltda, 2011. Cap. 2. p. 16-32.

[10]FUKS, Hugo; RAPOSO, Alberto Barbosa; GEROSA, Marco Aurélio. **Do Modelo de Colaboração 3C à Engenharia de Groupware**. WEBMIDIA 2003 - Simpósio Brasileiro de Sistemas Multimídia e Web, Trilha especial de Trabalho Cooperativo Assistido por Computador, Novembro 2003, Salvador-BA, pp. 445-452.

[11]GEROSA, Marco Aurélio. **Desenvolvimento de Groupware Componentizado com Base no Modelo**

3C de Colaboração. 2006. 252 f. Tese (Doutorado) - Curso de Informática, Departamento de Informmática do Centro Técnico Científico da Puc-rio, Puc-rio, Rio de Janeiro, 2006.

[12]GEROSA, Marco A.; FUKS, Hugo; LUCENA, Carlos J. P. **Elementos de percepção como forma de facilitar a colaboração em cursos via internet**. XII Simpósio Brasileiro de Informática na Educação - SBIE 2001. Vitória, 2001.

[13]HERNANDEZ, Emerson Barbiero. **Macanudo**: uma abordagem baseada em componentes voltada a reuso de projetos de hardware. Dissertação (Mestrado). Universidade Federal do Rio Grande do Sul, Instituto de Informática. 2005.

[14]PIMENTEL, M., GEROSA, M.A., FILIPPO, D., RAPOSO, A., FUKS, H. & LUCENA, C.J.P. **Modelo 3C de Colaboração no Desenvolvimento de Sistemas Colaborativos**. Anais do III Simposio Brasileiro de Sistemas Colaborativos, Natal - RN, 20 a 22 de Novembro de 2006. ISBN 85-7669-097-7. Porto Alegre: SBC, 2006. pp. 58-67.

[15]SANTOS, Vaninha Vieira. **Gerenciamento de contexto em sistemas colaborativos.** Tese (Doutorado). Universidade Federal de Pernambuco, Centro de Informática, 2006.

[16]SZYPERSKI, Clemens. **Component Software**: Beyond Object-Oriented Programming. 2ed. London: Addison-Wesley, 2002. ISBN 0-321-75302-X.

[17]TAROUCO, L. M. R., KONRATH, M. L. P., ROLAND, L. C. O professor como Desenvolvedor de seus Próprios Jogos Educacionais: Até Onde Isso é Possível? In: XV Simpósio Brasileiro de Informática na Educação (2004).