# GorimWeb: Role-Playing Game for Hydric Resources Management on Web Platform

**Vinícius Martins[1], Giovani Farias[1], Marina Fernandes[1], Raquel Barbosa[2], Eder Gonçalves[1], Diana Adamatti[1]**

[1]Federal University of Rio Grande
Center for Computational Sciences, RS, Brazil

[2]Federal Institute of Education, Science and Technology of Rio Grande do Sul
Rio Grande, RS, Brazil

giovanifarias@gmail.com

***Abstract.*** *Computational games have been increasingly used not only for entertainment but also for corporate training. Role-playing games are particularly well-suited for training purposes. This study focuses on the implementation and analysis of a web-based role-playing game for hydric resources called GorimWeb. For the implementation, various requirements were defined. These included supporting the game board's playability, storing and exporting data such as logs and chat, accommodating varying numbers of players, and integrating multi-agent systems. Additionally, the game needed a well-structured user interface, good performance, and reliable web availability. The software architecture of the game was also defined. Validation was conducted through two surveys, one before and one after gameplay. The results allowed us to assess the interfaces and hypotheses using the survey responses and the messages exchanged via chat during the game.*

## 1. Introduction

*Natural Resources Management* (NRM) is a field dedicated to addressing the problem of ensuring high-quality resources for the future by managing and improving the quality of today's resources. This involves overseeing nature reserves, private and public properties, water resources (such as watersheds and oceans), plant life (including forests and urban greenery), and animal life (both wild and domestic). However, NRM faces three primary challenges: (I) optimization and control; (II) management and communication; and (III) data analysis. To address these challenges, there is a need for specialized tools and strategies [Fuller et al. 2007].

*Artificial Intelligence* (AI), particularly *Multi-Agent Systems* (MAS), utilizes intelligent entities known as agents. These agents use information about their environment to independently and heterogeneously decide the best plan to achieve personal goals, which collectively contribute to the system's overall objectives [Lesser 1999]. When multi-agent systems are applied to simulations, they are referred to as *Multi-Agent Based Simulations* (MABS). MABS aims to simulate environments where agents interact with each other to find or predict solutions to problems, often in an interdisciplinary manner [Gilbert and Troitzsch 2005, Le Page et al. 2016]. The interdisciplinary nature and the tools provided by MABS make them highly suitable for designing and simulating problems within the scope of natural resources management.

Another tool that has been used in this field is the *Role-Playing Game* (RPG) [Perrotton et al. 2017]. In RPGs, players receive roles (characters) at the beginning of the game and perform these roles until the end. These games are usually divided into turns, allowing all players to make their own actions, with each action influencing the environment before the next turn. RPGs can be played using printed materials (with boards/cards), orally (with a storyteller and verbal choices), or computationally. Some projects [Adamatti et al. 2009, Le Page et al. 2016, Le Page and Perrotton 2017] have utilized RPGs to solve problems for workgroups by having players interpret the roles of coworkers in the game. This approach helps players understand their coworkers' needs and the reasoning behind their choices. Consequently, in the field of NRM, RPGs can be valuable tools for aiding decision-making among entities such as committees.

The main goal of this paper is to present GorimWeb[1], an RPG (Role-Playing Game) for Hydric Resource Management on the Web, and a decision support tool designed to assist the local management committee in its decision-making process. Section 2 outlines the theoretical foundation of this work. Section 3 describes the development of the RPG, starting with a tabletop version and progressing to a web version. Section 4 presents the testing of GorimWeb and its results. Finally, Section 5 provides the conclusions and discusses potential future work.

## 2. Theoretical Backgroung

### 2.1. Multi-Agent Systems

Multi-agent systems are systems in which many agents work independently to achieve their own goals, which collectively contribute to the main goals of the system. Agents within MAS can be homogeneous (learning in the same way) or heterogeneous (learning in different ways), but in both cases, they operate asynchronously from each other's threads [Lesser 1999]. Despite their operation, agents must be capable of organized actions, as most issues they address are solved in a distributed manner. Therefore, MAS must incorporate characteristics such as cooperation, coordination, competition, negotiation, and communication [Bordini et al. 2001, Bousquet and Le Page 2004]. Multi-agent systems can be advantageous in several ways. Firstly, they offer increased speed in problem-solving due to the distributed execution of agents' activities. Secondly, Multi-agent systems provide flexibility and scalability by connecting multiple systems. Lastly, they offer a broader scope of the information the system returns, as all resources operate within the same environment [Leitzke et al. 2019].

### 2.2. Role-Playing Games

A type of game is called a role-playing game, which is situated between the fields of gaming and theater because it combines player interaction with the acting of characters. Players act within an environment with predefined rules and behaviors to complete a given story [Pereira 2003]. In this way, most role-playing games do not have winners or losers, making them a collaborative type of game that reveals aspects of social relations and allows for the direct observation of interactions between players [Barreteau et al. 2003, Adamatti et al. 2009]. RPGs are constantly used in training because of their approach of simulating situations without real-world consequences, while

[1]GORIM stands for sãoGOnçalo river basin and miRIM lagoon, located in the South of Brazil.

still providing valuable learning experiences. This makes RPGs a playful tool that facilitates the learning of various subjects [Perrotton et al. 2017]. RPGs and MAS are very similar [Barreteau et al. 2001]: agents can be understood as roles, the environment as the board, and the steps of the system as the game rounds, making them suitable tools.

### 2.3. Hydric Resources Management

Water is the most vital natural resource for the ecosystem and its populations due to its essential role in the life and health of plants and animals, as well as its importance in social and economic human matters [Ponte et al. 2016]. The field of natural resources management studies the methods of managing water, animals, plants, and land, with a focus on maintaining environmental quality for future generations. One of the areas of NRM is *Hydric Resources Management* (HRM), which discusses the best usage and distribution of water, involving organizations and different social groups. In addition, HRM is responsible for addressing social and environmental risk situations, such as floods and droughts, that require preventive actions by authorities [Tundisi 2006]. Watersheds are geographical areas where, according to their relief, rainwater or water from sources drains to the bottom of the valley, exiting through a single point (usually a river). If there is a crest along the way, the water is divided between two different watersheds [Ahmad and Simonovic 2005].

### 2.4. Related Works

In [Farias et al. 2019], a systematic literature review was conducted following the methodology proposed in [Mariano et al. 2017]. The review aimed to consolidate research from the main databases across three primary fields: RPG (Role-Playing Games), MAS (Multi-Agent Systems), and NRM (Natural Resource Management). The review identified 10 articles, which were classified into 4 groups:

- **RPG → MAS**: this group includes papers where the RPG is played by stakeholders using cards or paper, and at the end of the game, the resulting information is converted into data for a multi-agent based simulation.
- **MAS → RPG**: projects in this group involve researchers developing a MAS first, followed by an RPG that uses the same metrics as the MAS, which is then validated by the stakeholders.
- **RPG + MAS**: the systems in this group feature a paper-based RPG complemented by a computational calculation module to handle complex calculations. Thus, the RPG and the MAS function together.
- **RPG ++ MAS**: this group includes projects where both the RPG and the MAS are computationally integrated and operate together.

The articles of the last two groups were selected as related projects due to their relevance to the current state of the project (RPG + MAS) and its ultimate goal (RPG ++ MAS). Some of these articles are discussed below.

In [Campo et al. 2009], an environment is developed to facilitate communication and negotiation among various stakeholders about their interests, the consequences of strategies, and the identification of intervention areas. The project proposed in [Le Page and Perrotton 2017] introduces a new approach to environmental modeling. To test their method, the authors created a MAS and an RPG aimed at aiding the decision-making process of a village facing potential incidents in plantations.

In [Adamatti et al. 2009], a project is described where the authors developed a tool for players to determine the quality and quantity of water in a capture system within an urban perimeter. Two prototypes were created: in the first, participants played with other participants, and in the second, participants could play randomly with intelligent agents.

## 3. Gorim

Gorim is a game designed to serve as a decision support tool for a local management committee. In this RPG, players take on the roles of various committee members, including `businesspeople`, `farmers`, `environmental inspectors`, `NGO`, `mayors`, and `councilors`. Due to the complexity and volume of calculations required (percentages, sales, productivity, etc.), we defined a game engine (see Figure 1).

### 3.1. Game Structure

The game begins with roles being randomly assigned to the players. There are five playable character classes: `businessperson`, `farmer`, `environmental inspector`, `mayor`, and `councilor`. These roles are distributed between the two cities in the game: Atlantis and Cidadela. Each round of the game is divided into two phases. In the first phase, the `businesspeople` and `farmers` play, and in the second phase, the `inspectors`, `mayors`, and `councilors` play.

The `businesspeople` can be of four types: *seed*, *fertilizer*, *machinery*, or *agrochemical*. They are responsible for negotiating the price of their products with the `farmers`. The `seed businessperson` has the option to sell *vegetable*, *rice*, or *soybean* seeds. The `fertilizer` and `agrochemical businesspeople` have three types of their products: *common*, *premium*, and *super premium*. The `machinery businessperson` can rent *machinery packages 1*, *2*, or *3*, as well as rent a *sprayer*. All sales generate specific **pollution** from the product (simulating its manufacture), and all sales revenue returns to the `businessperson` as **productivity**, which will be the additional amount of "money" available to the `businessperson` in the next round. Furthermore, the `businessperson` can sell products at three different prices: *low*, *normal*, and *high*, and can choose different prices for each client.

The `farmer` has six plots of land and can choose to plant on all or some of them in each round. In each plot, it is possible to use one type of seed, one type of fertilizer, and one type of agrochemical or machinery package. Each combination of seed, fertilizer, and agrochemical or machinery generates different levels of productivity and pollution for the `farmer`. Each plot can also receive the "green seal", awarded by the city's `environmental inspector` when the `farmer` does not use agrochemicals on that plot. Having the green seal results in a 5% discount on that plot's productivity tax.

At the end of the first phase of each round, these two classes – `businesspeople` and `farmer` – must pay taxes to their city hall. At the beginning of the game, the players are divided between cities, with the `seed` and `machinery businesspeople` in Atlantis and the `fertilizer` and `agrochemical businesspeople` in Cidadela. This division ensures a balance in the taxes received by each city. In addition, the roles for the second phase are played by the same players as in the first phase, with elections held every two rounds.

The `environmental inspectors` are responsible for analyzing the activities of the actors from the first phase belonging to their city. For those who have excessively polluted, the `inspector` can impose a *light*, *medium*, or *heavy fine*, or *no fine* at all, depending on the situation and the agreement that can be made with the other player. In addition, they are responsible for awarding or removing the green seal from the `farmers`' plots. All money received from fines is directed to the local city hall.

The `mayors` must also analyze the actions within their city, with the goal of controlling global pollution. This actor can implement three different environmental actions: *water treatment*, *waste management*, and *sewage control*, each with a specific cost and impact on global pollution. Additionally, the `mayor` has the power to adjust tax rates, which are divided into three classes, offering different tax rates for various productivity ranges. They must also consider management ideas proposed by their `councilors`.

The `councilor` of each city must assist its `mayor` with city management by analyzing the impacts of actions taken and money invested, as well as by convincing the `mayor` of their ideas. In addition, they must listen to the citizens and convey relevant information to the `mayor`.

The board game requires a minimum of six players, four of whom are `businesspeople` and the rest `farmers`, divided between the two cities. Every two rounds, three players in each city are chosen for the elective positions (`inspector`, `mayor`, and `councilor`), participating in both phases of the round. The remaining players – if there are more than six – play only the first phase and influence the second by questioning their representatives. Besides, all roles are allowed to transfer "money" to each other. The game ends when global pollution reaches 100%. Rising pollution decreases players' productivity by simulating soil infertility and water and air impurities for farming. Furthermore, the cost of the `mayors`' environmental actions increases, simulating the higher expense of removing greater amounts of pollution.

Although the game starts with rounded amounts of "money" for each player, the process of manually calculating taxes, purchases, sales, and global and individual pollution has proven impractical. This highlighted the need for a computational calculation module. The calculation engine was programmed in JAVA using agent-oriented programming. At the end of each stage, the engine produces summary files for the players, detailing their actions in the last stage. Additionally, throughout the game, log files are created to store all game actions.

### 3.2. API Web

The development of GorimWeb began with the back-end for two primary reasons: (1) the completion of the game engine, shown in Figure 1, which is a crucial component of the project's structure, and (2) the necessity of having the API operational to enable testing and streamline the development of the components for the players' interfaces.

Figure 1 illustrates the layered architecture proposed for the application's back-end. The API receives JSON requests from players, each following a specific format defined in the models. The API then converts these JSON requests into the designated formats and transmits them to the `WorldService`. At the start of the game, `WorldService` initializes the objects to be used, interfacing directly with the `Game Engine`. As player actions are received by `WorldService`, it invokes the necessary

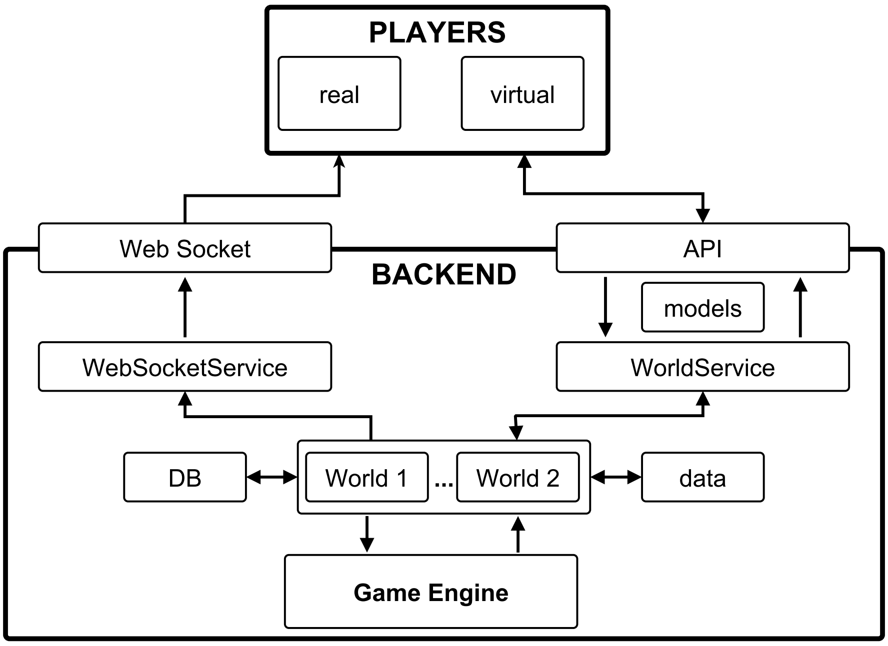


**Figure 1. Conceptual view of the back-end component architecture.**

methods within each class of the `Game Engine`. All actions are consistently recorded in the database (`DB`), and at the end of each stage, summary files are generated and stored within the `data` component. These files are saved in JSON format and are subsequently dispatched to players upon request.

The back-end was implemented using JAVA Spring Boot. The `Game Engine`, shown in Figure 1, is exactly the same as the one used in Gorim matches. Additionally, a web socket was implemented, which allows the use of chat through the interface, where all messages are saved in the database (`DB`), and players can send suggestions, budgets, and responses.

### 3.3. Player's Interfaces

The implementation of the graphical interface, developed using Angular, was entirely divided into modules, with each possible window of the interface being a *module*. Each *module* is separated into *component* (processing class), *service*, *model*, *interface* (HTML, SCSS or CSS files), and *subcomponents*. These are substructures that have been separated from the main class of the *module* because they have their own behavior or different calls to the back-end. Each of them has its own *service*, *interface*, *component*, and sometimes *subcomponent* and *model*.

The *services* are used to acquire and send data to/from the back-end, with a *component* as the origin and destination of these data. Another use of *services* is to transfer data from one *component* to another, keeping an open channel for this internal exchange. For example, when an `environmental inspector` marks that they fined a `businessman`, internally, an *component* sends this "fine" to the main *component* of the supervisor *module*, where it is stored until the player cancels this fine or the end of the stage. At the end of the stage, using the main *service* of the *module*, this list (and the green seal list) is sent to the back-end.

Figure 2 shows the component diagram of how the project front-end operates. MODULE represents any of the implemented modules, except for WORLD, which are subcomponents common to others. Therefore, when a page (e.g., `farmer`) is accessed, the main class of the MODULE initializes the necessary SUBCOMPONENTS – including those lodged in WORLD – for the page's functionality. Besides the main modules, such as `businessman` and `farmer`, there are also auxiliary modules, such as ALTERNATIVE-SECOND-PAGE (an alternative window for players not playing an elected role in the second stage, allowing them to use the chat and converse with other players, ask questions to city managers, etc.); HOME (the initial game window where one can join a game as a player or create a new game as a master); GAME-OVER (the final game window showing a summary of the players); MASTER (the window where the master controls the game, starting new stages or ending the game); and WORLD.

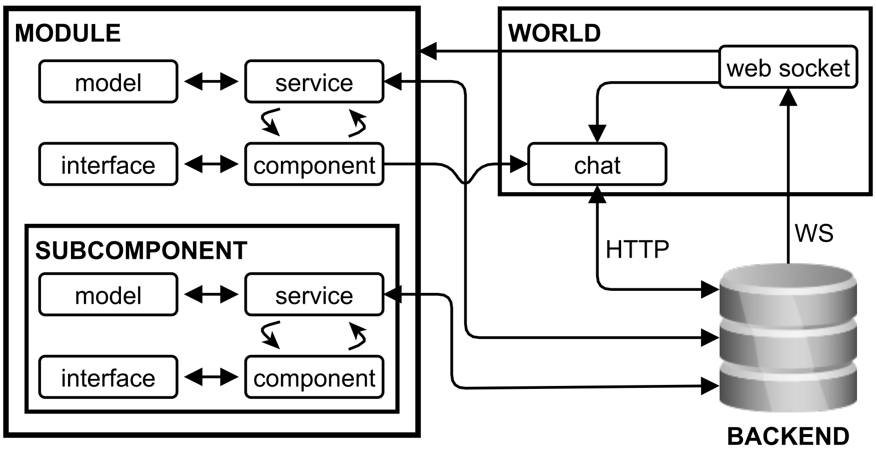


**Figure 2. Component diagram of the *front-end*.**

With these components, the graphical interface for the players was implemented. For example, a farmer's interface is shown in Figure 3, which includes: (1) a component displaying current information about the player and the environment; (2) a button to activate the side tab showing the quotes sent to the farmer; (3) a menu to display the installment form or value tables; (4) a form for sending money; (5) a representation of the parcel – land plot – form; (6) a chat for exchanging information between players.

## 4. Tests and Results

This Section presents the results of the test games conducted, based on the questionnaires applied to the GorimWeb players, as well as the presentation of the analyzed group.

To assess the functional and non-functional requirements of the architecture, as well as the ease of navigation of the character interfaces, the questions listed in Tables 1 (pre-game form) and 2 (post-game form) were asked. These tables also show the types of responses: open responses, responses with presented options (such as yes and no), and responses using the Likert scale (where a scale of 0 to 5 is used, with 0 being 'completely disagree' and 5 'completely agree').

The group of people who tested the game had never played Gorim before, although two of the six players had a general understanding of how the game worked. The game

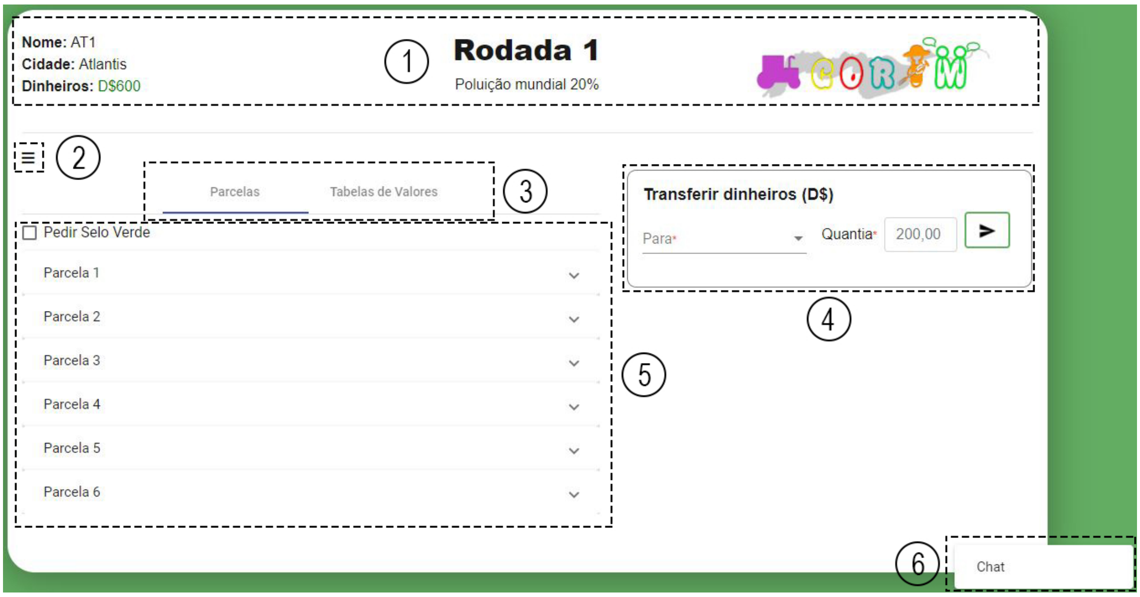


**Figure 3. Farmer's interface.**

**Table 1. Pre-game form questions.**

| # | Questions | Response |
|---|---|---|
| 1 | What is your name? | open response. |
| 2 | Have you ever played an RPG? | yes or no. |
| | **If yes** | |
| 2.1 | Was this RPG Gorim? | yes or no. |
| | Which mode of Gorim was it? | in-person, remote, or both? |
| 2.2 | What features do you think are essential in an online RPG, specifically, if it were set in the world of Gorim? | open response. |

session involved six people and lasted two hours (a total of 7200 seconds). During the session, players sent 660 messages, with 390 of those messages exchanged with the game master (196 sent by the game master and 194 received from the players).

In Figure 4, we have a graphic constructed with the time of the game in seconds on the horizontal axis (X-axis) and the chat rooms opened during the games on the vertical axis (Y-axis). The number of chat rooms created varies from game to game because players may or may not choose to contact each other. For instance, in one game, farmer AT1 might contact farmer CD2 to ask about the seeds he is using, while in another game, this may not happen. In this situation, game 1 would have a room AT11-CD21, while game 2 would not since rooms are only created if at least one message is sent between two players. Therefore, points marked on the graph represent messages sent by players – including the game master – over time, in seconds, from the game start. Additionally, five vertical lines on the graph represent the transition between stages of the game. The highlighted horizontal area shows messages sent or received by the game master. The chat rooms are organized so that messages exchanged by any player during the first stage are below the messages exchanged with the game master, and those above are messages

**Table 2. Post-game form questions.**

| # | Questions | Response |
|---|---|---|
| 1 | What is your name? | open response. |
| 2 | What was your character in the first stage? | `businessperson` or `farmer`. |
| 3 | And in the second stage? (You can mark more than one if you were elected to another role) | "`env. inspector`", "`mayor`", "`councilor`", or "no role". |
| 4 | Did all the features you expected to be in the game appear? | yes or no. |
| 5 | Did you find all the information related to your character? | yes or no. |
| 6 | Did you find all the necessary information about other players for your gameplay? (e.g., names, product values, etc.) | yes or no. |
| 7 | Did the message exchange (chat) provide the necessary interaction with other players? | yes or no. |
| 8 | Was the message exchange quick? | Likert scale. |
| 9 | The game was not slow. | Likert scale. |
| 10 | During the game, there were no delays in data processing. | Likert scale. |
| 11 | Information about your own character was easily found. | Likert scale. |
| 12 | Necessary information about other characters was easily found. | Likert scale. |
| 13 | In the pre-game questionnaire, did you mention you had played Gorim before? | yes or no. |
| 13.1 | **If you marked yes** Did you find the GorimWEB interface similar to the tabletop Gorim game (paper sheets)? | yes or no. |
| 13.2 | If not, what changed? | open response. |
| 13.3 | Were the game's objectives clear? | yes or no. |
| 13.4 | Did the game contribute to your understanding of WRM? | yes or no. |
| 14 | Finally, if you would like to make any comments (something I did not ask that you think is important, praise, criticism). | open response |

exchanged with players during the second stage.

Figure 4 presents a graph that illustrates the distribution of messages for this group. In this group, 27 chat rooms were opened and the message distribution and duration of each stage were as follows (where **R** stands for **Round** and **E** stands for **Stage**):

- **R1E1**: this stage lasted approximately 35 minutes (ending at the 2110-second mark), during which 233 messages were sent. Of these, 118 messages involved

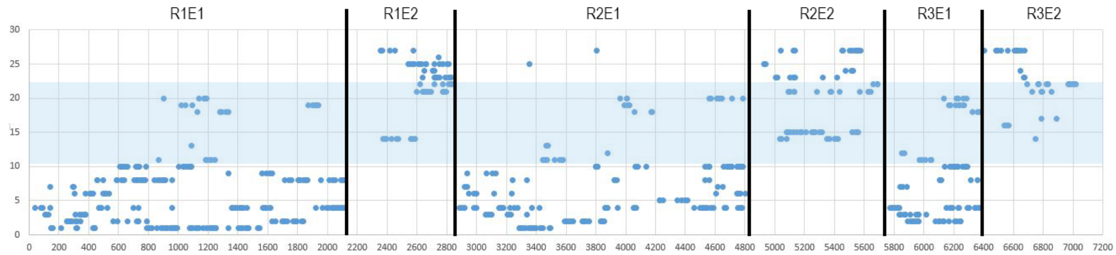


**Figure 4. Messages dispersion over time (in seconds) from the group start.**

the game master (56 sent and 62 received).

- **R1E2**: this stage lasted approximately 12 minutes (ending at the 2836-second mark), during which 62 messages were sent. Of these, 47 messages involved the game master (22 sent and 25 received).
- **R2E1**: this stage lasted approximately 33 minutes (ending at the 4801-second mark), during which 168 messages were sent. Of these, 93 messages involved the game master (45 sent and 48 received).
- **R2E2**: this stage lasted approximately 15 minutes (ending at the 5691-second mark), during which 65 messages were sent. Of these, 44 messages involved the game master (23 sent and 21 received).
- **R3E1**: this stage lasted approximately 11 minutes (ending at the 6372-second mark), during which 95 messages were sent. Of these, 56 messages involved the game master (30 sent and 26 received).
- **R3E2**: this stage lasted approximately 11 minutes (ending at the 7018-second mark), during which 37 messages were sent. Of these, 32 messages involved the game master (20 sent and 12 received).

## 5. Conclusion

This paper presented the theoretical foundation, related projects, and the current state of a project that intersects multi-agent systems, role-playing games, and natural resources management. The goal of this project was to develop a computational RPG to assist the decision-making process of the Mirim lagoon and the São Gonçalo river committee. Currently, the web-based game is complete and available online at the Federal University of Rio Grande server. It is now capable of being played by individuals who are not in the same location (link will be available if accepted). Additionally, as further work, we will conduct data mining on the simulation logs to analyze player strategies. This analysis will be used to develop non-player character agents, which will participate in the game when there are not enough human players to start a match.

## 6. Acknowledgements

This research was supported by Edital FAPERGS/CNPq 07/2022 – PDJ. We also thank *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES) and *Agência Nacional das Águas* (ANA) for the financial support of project number 16/2017.

## References


Adamatti, D. F., Sichman, J. S. a., and Coelho, H. (2009). An analysis of the insertion of virtual players in gmabs methodology using the vip-jogoman prototype. *Journal of Artificial Societies and Social Simulation*, 12(3):7.

Ahmad, S. and Simonovic, S. P. (2005). An artificial neural network model for generating hydrograph from hydro-meteorological parameters. *Journal of Hydrology*, 315(1-4):236–251.

Barreteau, O., Bousquet, F., and Attonaty, J.-M. (2001). Role-playing games for opening the black box of multi-agent systems: Method and lessons of its application to senegal river valley irrigated systems. *Journal of Artificial Societies and Social Simulation*, 4.

Barreteau, O., Le Page, C., and D'aquino, P. (2003). Role-playing games, models and negotiation processes. *Journal of Artificial Societies and Social Simulation*, 6(2).

Bordini, R. H., Vieira, R., and Moreira, A. F. (2001). Fundamentos de sistemas multiagentes. In *Anais do XXI Congresso da Sociedade Brasileira de Computação (SBC2001)*, volume 2, pages 3–41.

Bousquet, F. and Le Page, C. (2004). Multi-agent simulations and ecosystem management: a review. *Ecological modelling*, 176(3-4):313–332.

Campo, P. C., Mendoza, G. A., Guizol, P., Villanueva, T. R., and Bousquet, F. (2009). Exploring management strategies for community-based forests using multi-agent systems: A case study in palawan, philippines. *Journal of Environmental Management*, 90(11):3607 – 3615.

Farias, G., Leitzke, B., Born, M., Aguiar, M., and Adamatti, D. (2019). Systematic review of natural resource management using multiagent systems and role-playing games. In *18th Mexican International Conference on Artificial Intelligence - MICAI*, Xalapa. Springer.

Fuller, M. M., Wang, D., Gross, L. J., and Berry, M. W. (2007). Computational science for natural resource management. *Computing in Science & Engineering*, 9(4):40.

Gilbert, N. and Troitzsch, K. (2005). *Simulation for the social scientist*. McGraw-Hill Education (UK).

Le Page, C., Dray, A., Perez, P., and Garcia, C. (2016). Exploring how knowledge and communication influence natural resources management with rehab. *Simulation & Gaming*, 47(2):257–284.

Le Page, C. and Perrotton, A. (2017). Kilt: A modelling approach based on participatory agent-based simulation of stylized socio-ecosystems to stimulate social learning with local stakeholders. In Sukthankar, G. and Rodriguez-Aguilar, J. A., editors, *Autonomous Agents and Multiagent Systems*, pages 31–44, Cham. Springer International Publishing.

Leitzke, B., Farias, G., Melo, M., Gonçalves, M., Born, M., Rodrigues, P., Martins, V., Barbosa, R., Aguiar, M., and Adamatti, D. (2019). Sistema multiagente para gestão de recursos hídricos: Modelagem da bacia do são gonçalo e da lagoa mirim. In *Anais do X Workshop de Computação Aplicada a Gestão do Meio Ambiente e Recursos Naturais*, Belém, Pará, Brasil.

Lesser, V. R. (1999). Cooperative multiagent systems: a personal view of the state of the art. *IEEE Transactions on Knowledge and Data Engineering*, 11(1):133–142.

Mariano, D. C. B., Leite, C., Santos, L. H. S., Rocha, R. E. O., and de Melo-Minardi, R. C. (2017). A guide to performing systematic literature reviews in bioinformatics. *arXiv preprint arXiv:1707.05813*.

Pereira, C. E. K. (2003). Construção de personagem & aquisição de linguagem: O desafio do rpg no ines. In *vol. 10,(jul/dez) Rio de Janeiro INES, 2004 Semestral ISSN 1518-2509 1–Forum–Instituto Nacional de Educação de Surdos*, page 7.

Perrotton, A., Garine-Wichatitsky, D., Fox, H. V., and Le Page, C. (2017). My cattle and your park: codesigning a role-playing game with rural communities to promote multistakeholder dialogue at the edge of protected areas. *Ecology and Society*, 22(1).

Ponte, B., De la Fuente, D., Parreño, J., and Pino, R. (2016). Intelligent decision support system for real-time water demand management. *International Journal of Computational Intelligence Systems*, 9(1):168–183.

Tundisi, J. G. (2006). Novas perspectivas para a gestão de recursos hídricos. *Revista USP*, 1(70):24–35.