



e-Game Technical Development

This document describes the technical aspects of the eGame developed in the WP4 of the project.



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

TURKU AMK
TURKU UNIVERSITY OF
APPLIED SCIENCES



1862

RĪGAS TEHNISKĀ
UNIVERSITĀTE



**FACHHOCHSCHULE
WIENER NEUSTADT**
University of Applied Sciences – Austria



Funded by
the European Union

Project Data

Project acronym	RESPRO
Project name	Developing people skills in project management education in European HEI's
Project code	2022-1-ES01-KA220-HED-000085940

Document Information

Title	e-Game Technical Development
Work Package	4
Dissemination Status	Public
Authors/Contributors	Lucas Miakinen (RTU), Jose-Luis Poza-Lujan (UPV),

Document History

Version	Date	Changes and comments

Disclaimer: *This document was co-funded by the European Union. Its contents are the sole responsibility of authors and do not necessarily reflect the views of the European Union. Co-funded by the European Union or of other members of the RESPRO consortium.*

Table of contents

EXECUTIVE SUMMARY 5

INTRODUCTION 5

About the project 5

 Mission 5

 Vision 5

 Values..... 5

About this document 6

USE CASES..... 6

Roles..... 6

 Players..... 6

 Teachers..... 6

 System Administrators/Technicians 7

Cases 7

 Case 1. Scenario creation 7

 Case 2. Playing the game 8

 Case 3. Teachers taking the results..... 9

TECHNICAL FRAMEWORK 10

System architecture & technology stack 10

COMPONENT INTERACTIONS..... 10

 Scenario creation (teacher, administrator)..... 10

 Gameplay (student) 11

 Results & feedback (teacher)..... 11

IMPLEMENTATION 11

Considerations..... 11

 Overview of the game in the context of the project..... 11

 Technical Considerations in Browser-Based Gaming..... 12

Gameplay elements 12

 Workers..... 12

 Rooms 12

 Open Space 13

 Lounge 13

 Library 13

 Game Cycles..... 14

 Side Missions..... 14



Game Lifetime.....	14
Game Design.....	14
User interface (UI).....	14
3D Rendering	15
Game Environment	15
Board.....	15
Rooms	15
Room Walls	15
Workers.....	16
Technical Specifications	16
3D rendering optimization.....	16
Optimized cross-platform rendering.....	17
Flow Diagrams	17
Technical Working.....	17
Game Engine System Diagram	18
Game User Interface (UI) Diagram.....	18
Game Event System Diagram.....	19
CONCLUSIONS	20



Executive summary

The RESPRO project aims to enhance project management education by developing people skills and focusing on responsible leadership and well-being at work. With the rise of international projects and the increasing complexity of team dynamics, the need for project managers to possess strong leadership and interpersonal skills has become more critical. The project addresses these needs by creating innovative learning tools: easy-access short courses and an e-simulation game focusing on responsible leadership components. These tools promote team well-being with proactive leadership, and using job crafting, responding to Europe's growing demand for soft skills in project management education. Led by a consortium of four higher education institutions from Spain, Finland, Latvia, and Austria, RESPRO leverages interdisciplinary expertise to ensure the relevance and sustainability of its outcomes. Through transnational collaboration and integrating innovative pedagogical approaches, RESPRO aims to strengthen project managers' ability to lead effectively in diverse and complex environments, fostering a more resilient and people-focused approach to project management.

Introduction

About the project

Mission

The mission of the RESPRO project is to reinforce the people skills in project management education across European HEIs by analysing best practices in responsible leadership, integrating these principles into curricula, and delivering Easy-Accessible Short (EAS) courses, an e-simulation game, and a Teacher Guidebook that can help HEIs institutions to create these skills in their students.

Vision

The vision that guides the project is create a higher-education ecosystem where responsible leadership and well-being at work are embedded as core competences of every project manager; where educators and students use job crafting as a method to achieve this ecosystem; and where RESPRO's open resources are used as a model of tools to use.

Values

The RESPRO project is guided by a series of values that must be followed to ensure a framework of understanding that is both caring and efficient. These values are as follows:

- Responsibility & Ethics. We consider responsibility as the knowledge of the consequences of decisions.
- Well-being & Psychological Safety – We foreground team well-being as a driver of progress.
- Openness – We commit to open access for results and materials, enabling broad adoption and reuse.
- Inclusion & Accessibility – We design for diverse learners and contexts, ensuring activities and outputs are accessible to all.
- Evidence-based Improvement – We ground decisions in research and evidence to refine processes and outputs.
- Co-creation – We build with and for our communities: students, teachers, practitioners, and associated partners.

- Digital Readiness – We deliver practical, user-friendly digital solutions that enhance teaching and learning.
- Impact Orientation – We focus on measurable value for learners, educators, institutions, and the labour market, and on the transferability of results.

About this document

This document presents the technical development of the RESPRO e-simulation game (Result R4.2) within Work Package 4 e-Simulation Game. It provides a transparent record of the decisions, designs, builds and validations that led to a releasable tool. The document covers the work carried out in T4.2 (Technical development) including design, architecture, and implementation approach. It is scoped to the functionality planned for the technical phase.

Use cases

Use cases are used to describe how the software behaves based on the role of each of the possible roles. These roles are simplified, having only three types to explain and develop and focus more on pedagogical actions. Based on the roles, the use cases in which the application has been divided will be presented to clearly differentiate the options that a teacher should have from those that a student should have. All these roles are presented below.

Roles

According to the evaluation and analysis made by the project partners, it was determined that there would be three types of roles.





St 	PM 	Te 	SA 
Students and project managers (role: player)	Teachers	Administrators	

Figure 1. Symbols used to the profiles of eGame users and their role for the user cases.

These roles facilitate learning through contextualized issues in a scenario. The necessary skills to be a good responsible leader generate a good team environment and allow job crafting to be the most appropriate method for these objectives.

Players

Players are the profile of those people who enter the game to be able to resolve issues in the scenarios, that is, those who want to learn or check their alignment with the transversal competencies necessary to develop responsible leadership. Some user profiles that fit this role are:

- (Master's) Students
- Project management students (as a main specialization).
- Engineering and business administration students including project management in their curriculum.
- Undergraduate (bachelor's level) students, since future professionals should be familiar with the study contents

Teachers

The role of teacher has been determined as the one who can create scenarios and determining the situations that will occur in the scenario. In addition, he/she must also be able to include the questions

that will be selected for each room of each scenario and the answers as well as the assessment. This role must also be able to review the results of each of the players to determine which skills they have already acquired, and which ones require improvement, which are determined to achieve the best practices of responsible leadership. Some user profiles that fit this role are:

- Project management professors in engineering and business administration programs.
- HEI (Higher Education Institutions) professors wishing to integrate RESPRO project results into their courses.
- Professionals in project management that want create training in their companies.

System Administrators/Technicians

The administrator role consists of the person who enters to be able to tune and adjust the application for the specific needs of each of its uses, that is, it is not a player or a teacher but the person who has the “full stack” access of the application.

Cases

Below we will describe the case studies in which the gameplay of the application is divided or determined. These case studies will be determined by the role that initiates each of the actions as well as the steps and elements that intervene in each one.

Case 1. Scenario creation

The user case 1 is the scenario creation. Scenarios are created by the role “Teacher”. These scenarios need the support of Project Manager Professionals to match the needs of the training with the scenario created.

In this case study, what we have is that the users who have permissions to create a scenario, which are the teachers and the administrators, must create the scenario through two of three steps. The first of them is to create a new company. If this company is already created, they must create the necessary rooms for that company. With these rooms, scenarios are created, and these scenarios involve a form that will be described later.

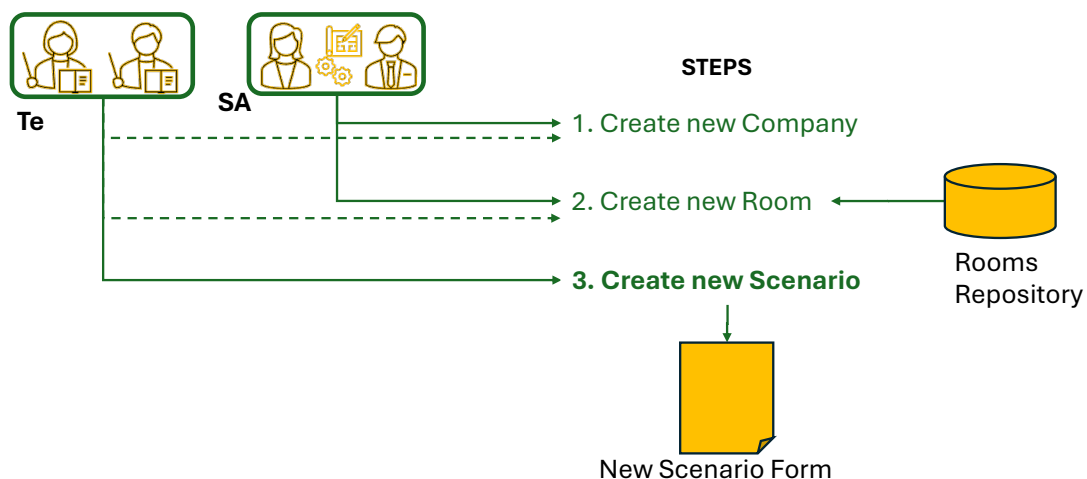


Figure 2. Use case of create scenarios (part 1 of 2).

To create a company, it is necessary to have the permissions of professor administrator and that company represents either a real environment that can be a real company part of the university or even a fictitious environment that is decided to use to simulate situations or do role-playing.

To create a room, it is necessary to have a company already created. The number of rooms that a company can have is unlimited, considering that these rooms represent environments in which a situation, part of the scenario, will develop. In other words, this room is there to set the simulation and the Player role.

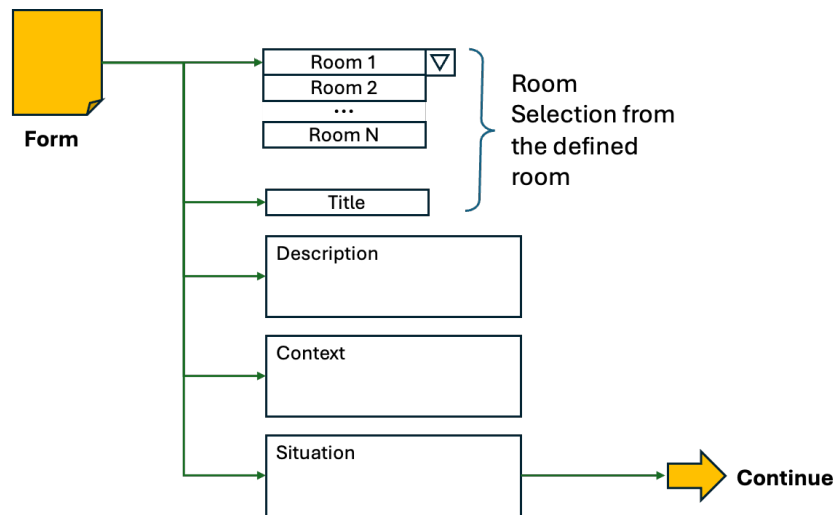


Figure 3. Data needs and flow of the user case of create scenarios (part 1 of 2).

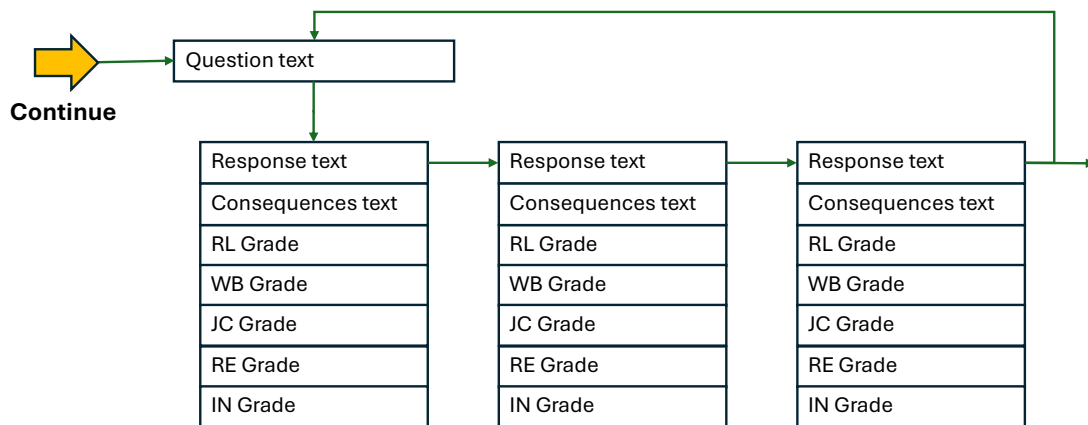


Figure 4. Data needs and flow of the user case of create scenarios (part 2 of 2).

The form to create a scenario consists of first selecting rooms that have already been defined or the part of returning to a new room as described in the scenario creation. Additionally, the context and the situation, can be detailed. A scenario is always associated with a room. From there, it is possible to configure the situation. Next, it is possible to create test questions. A test question has a set of answers that, although by default it is determined in three, that number of answers can be changed, meaning it has consequences because different answers have different consequences and then there is an assessment by Responsible Leadership Well-being, and Job Crafting and also some other parameter that is determined or that those who create the scenarios want to determine.

Case 2. Playing the game

The player case study has three steps, the first is obviously to enter a company and create a scenario within the company and from there within the scenario there is a route of rooms in which each of them has a series of questions about the scenario, that is to say that in a room different situations can occur, these situations are very important because they are the ones that will determine the player's answers that will give us the necessary clues to know what leadership skills he has and which ones he is lacking.

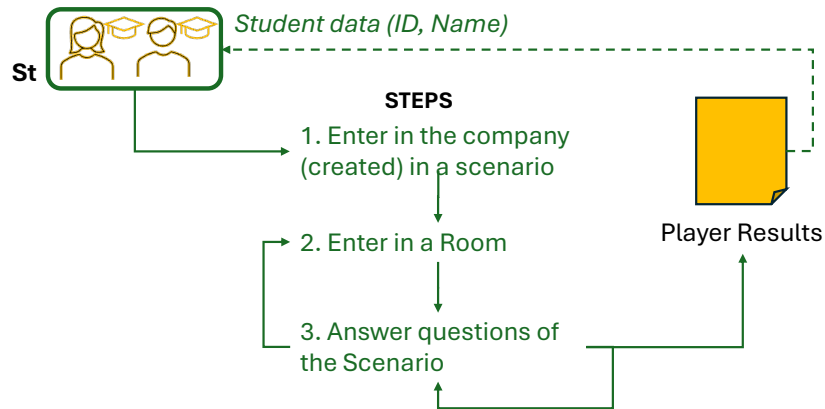


Figure 5. Data needs and flow of the user case of create scenarios (part 2 of 2).

The scenario in which the game has been played generates a series of reports with the answers to the questions that will also be evaluated by the formulas already discussed in previous chapters.

Case 3. Teachers taking the results

The third case involves obtaining player data. In this case, the teacher accesses the platform to review the results. In this case, the functions to be performed are first to select the player, that is, the user who has performed all the actions. From there, you can access their results, both the last ones and those previously performed. From there, you can read the results, and they can be fed back through comments or they can also be exported for external reports in PDF format.

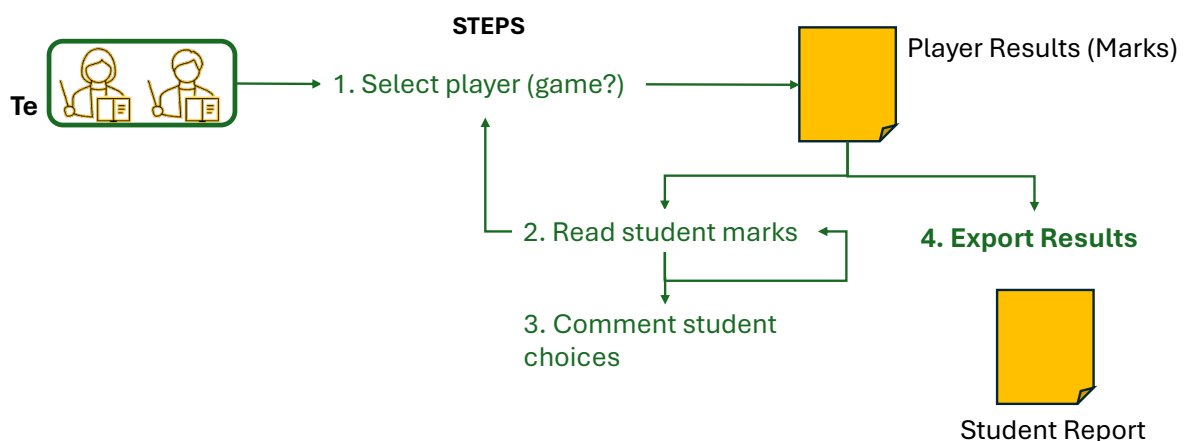


Figure 6. Data needs and flow of the user case of create scenarios (part 2 of 2).

These use case diagrams allow us to fully model all the actions you can do with the application in such a way that it covers the objectives determined by the partners in the previous work packages.

Technical Framework

This section implements the “System architecture & technology stack” item in the R4.2 e-Tool Technical Development deliverable outline, referencing the defined use cases and integration topics in that document. It also aligns with the WP4 technical development task and the results list where R4.2 Technical Development is specified

System architecture & technology stack

The RESPRO e-Simulation tool is designed to be modular, portable and easy to operate across different HEI environments. The architecture follows a service-oriented, container-based approach to:

- support the WP4 use cases (scenario creation, gameplay, results/analytics) with clear separation of concerns,
- integrate smoothly with institutional LMSs and SSO, and
- meet privacy, security, and maintainability requirements expected in HEI settings.

These goals align with the scope of R4.2 “Technical Development” and its companion sections on LMS/SSO integration and non-functional requirements. To achieve these goals, the system is organised into four logical layers; each deployed as one Docker container:

- Presentation layer:
 - Web Frontend: a browser-based SPA used by students and teachers for scenario authoring, game play, and reviewing outcomes.
 - Served behind a reverse proxy that also terminates TLS.
- Application layer
 - API Gateway exposing a stable REST/GraphQL surface to clients and partner systems.
 - Game/Simulation Service executing scenario logic, events, time progression and scoring.
 - Background Worker(s) handling long-running or asynchronous jobs (simulation ticks, report generation, LMS grade pass back).
- Data & analytics layer
 - Relational Database for core entities (users, scenarios, runs, telemetry snapshots).
 - Cache & Queue for low latency reads and job dispatch.
 - Object Storage for scenario media and exports.
 - Analytics / Reporting for teacher dashboards and exports.
- Integration & access layer
 - LTI Adapter for LMS launch, roster and grade return.
 - Identity/SSO via OpenID Connect (or SAML through a broker) to support campus logins.
 - Admin console for operational tasks.

These layers implement the three primary use cases specified in this deliverable: Scenario creation, Playing the game, and Teachers taking the results.

Component interactions

Scenario creation (teacher, administrator)

This flow maps to the “Scenario creation” use case.

- Teacher authenticates via SSO.

- Frontend calls API to create/edit a scenario (metadata, steps, events, assets).
- Media uploads are stored in Object Storage; references are persisted in PostgreSQL.
- Draft scenarios are validated by the Simulation Service (rules linting).
- On publish, a versioned snapshot is created; optional LMS deep-linking is prepared via the LTI Adapter for course embedding.

Gameplay (student)

This flow maps to the “Playing the game” use case

- Student launches from the LMS (LTI 1.3) or signs in via SSO; the LTI Adapter mints a login and hands context (course, role).
- Frontend loads the published scenario and opens a session.
- Simulation ticks run synchronously (API+Simulation service) for interactive steps and asynchronously (Worker) for time-based events.
- Snapshots and telemetry (choices, timings, outcomes) are written to PostgreSQL; hot state may be cached in Redis.
- Optional grade pass back is queued and sent to the LMS via the LTI Adapter.

Results & feedback (teacher)

- This flow maps to use case “teachers taking the results!
- Teacher opens the results dashboard.
- Aggregations are served from analytics views; heavy reports are generated by the Worker and stored in Object Storage.
- Feedback exports (CSV/PDF) are downloadable; selected scores can be returned to the LMS.

Implementation

Considerations

It is important to highlight that in the context of this implementation chapter the terminology will be slightly different from that used in the use cases and in the rest of the design carried out in the work package. This is because from the internal point of view of the implementation it is convenient to speak in another terminology because the variables and some of the aspects are very related to this last terminology.

Overview of the game in the context of the project

In the context of the Respro game, players are offered a 3D business management experience, navigating through the complexities of corporate strategy and resource management. Designed to simulate the challenges and opportunities of leading a prosperous business, Respro places players in the role of a corporate entity, inviting them to make strategic decisions to maximize financial gains. The game features an interactive 3D environment for strategic decision-making, ranging from the architectural construction of the company to the hiring of workers and undertaking company missions.

The primary objective of Respro is to make the company thrive by maximizing revenue. Different rooms and employee assignments contribute to varying income levels, creating a dynamic economic system influenced by players' strategic decisions. The game also presents challenges based on missions, featuring varying levels of risk and reward, adding an additional layer of strategy to overall business management. In the context of the Respro game, players are offered a 3D business

management experience, navigating through the complexities of corporate strategy and resource management. Designed to simulate the challenges and opportunities of leading a prosperous business, Respro places players in the role of a corporate entity, inviting them to make strategic decisions to maximize financial gains. The game features an interactive 3D environment for strategic decision-making, ranging from the architectural construction of the company to the hiring of workers and undertaking company missions.

The primary objective of Respro is to make the company thrive by maximizing revenue. Different rooms and employee assignments contribute to varying income levels, creating a dynamic economic system influenced by players' strategic decisions. The game also presents challenges based on missions, featuring varying levels of risk and reward, adding an additional layer of strategy to overall business management.

Technical Considerations in Browser-Based Gaming

Browser-based games face technical and performance limitations, particularly concerning graphical aspects and resource management. These constraints can impact the visual complexity of the game, and dependence on web technologies introduces considerations regarding browser compatibility. Potential adjustments may be necessary to ensure a consistent experience across various platforms.

While modern browsers are entirely capable of running games of this type, adjustments, especially on the graphical side, are often required. Achieving a photorealistic game on a browser while maintaining compatibility, smoothness, and optimization across different platforms remains a considerable challenge. Consequently, the game's visual approach adopts the "low-poly" strategy, simplifying the details of 3D models to make them lighter and less resource-intensive during display.

Gameplay elements

Workers

Employees play a central role in Respro, representing a vital aspect of the game's dynamics. They can be recruited with each possessing distinct skills that vary from one individual to another. These unique capabilities dictate their proficiency in specific tasks, impacting their overall effectiveness within the company.

However, the number of employees is limited, and it depends on the level of the company, which is determined by the generated revenue. As the financial success of the virtual corporation increases, the player gains the capacity to recruit and manage a more extensive and diverse workforce.

Effectively managing the recruitment and deployment of employees becomes a strategic consideration, as their skills directly contribute to the overall success and prosperity of the player's corporate entity in Respro.

Rooms

The game environment revolves around the strategic placement and management of various rooms within the 3D space. These rooms serve as integral components influencing the productivity and financial success of the player's corporate entity. The diverse range of rooms includes open spaces, lounge, and libraries, each contributing uniquely to the overall dynamics of the game.

Respro offers a selection of room types, each with distinct functions. Open spaces act as hubs for revenue generation, relaxation areas boost employee well-being, and libraries enhance intelligence levels.

Rooms have varying levels that directly influence their effectiveness. Upgrading the level of a room enhances its capabilities, whether it be generating more revenue, boosting employee skills, or improving overall satisfaction. This introduces an element of progression, encouraging players to invest in the continuous improvement of their corporate infrastructure.

The interactions between different room types and their influence on employee performance create a dynamic and evolving gameplay experience. For example, assigning employees with specific skills to appropriate rooms can amplify their contributions, ultimately impacting the bottom line of the corporation.

Open Space

In Respro, the Open Space is a pivotal room type that serves as a nucleus for revenue generation within the corporate structure. Strategically placing and optimizing Open Spaces is fundamental to the financial success of the player's virtual company. These areas act as collaborative hubs where employees work collectively, fostering an environment conducive to increased productivity and revenue.

Open Spaces are primary contributors to the overall revenue of the corporation. Employees assigned to these areas generate income based on their individual skills and qualifications. The strategic allocation of high-performing employees to Open Spaces enhances the revenue potential, making it a critical consideration for players aiming to maximize their financial gains.

The level of an Open Space directly affects its capacity to generate revenue. Upgrading the Open Space enhances its capabilities, attracting higher-skilled employees and, consequently, increasing the overall financial output. This introduces a strategic element, encouraging players to invest in the development of these critical hubs.

As the game progresses, players must adapt Open Spaces to changing dynamics, ensuring that they remain optimized for maximum revenue. This may involve adjusting the composition of employees, upgrading the room level.

Lounge

The Lounge in Respro serves as a vital room dedicated to enhancing the well-being index of employees, directly influencing their productivity and, consequently, the generated revenue.

Beyond its impact on well-being, the Lounge also contributes to the success of side missions. Indeed, the Lounge increases focus and effectiveness, thereby enhancing the likelihood of successfully completing additional missions. This dual functionality makes the Lounge a multifaceted asset in the strategic arsenal of the player. Like the open-space room, improving this room will increase its beneficial effects.

Library

The Library in Respro serves as a space dedicated to increasing the intelligence of workers. Like other rooms, upgrading it amplifies its effects. Intelligence, in turn, enhances both productivity and the likelihood of mission success among employees. Upgrading the Library becomes a strategic move to

boost overall workforce intelligence, thereby positively influencing the corporate productivity and mission outcomes.

Game Cycles

Respro operates in distinct daily cycles, where each in-game day corresponds to 8 hours (8 am to 4 pm), translating to 2 real-life minutes. At the conclusion of each day, players receive detailed feedback on their corporate performance. This provides players with frequent opportunities to assess and adjust their strategies based on daily outcomes. Importantly, if the company ends the day in deficit, the player loses.

Side Missions

In addition to the automatic revenue generation by workers, another avenue for both earning and potential losses lies in the realm of side missions. These missions, varying in complexity and risk, offer players the chance to further shape the financial trajectory of their virtual corporation. Successfully completing these missions contributes to the overall revenue and strategic growth of the company. However, players must exercise careful consideration, as failure in these missions can have repercussions, impacting the financial standing and progress of the corporation. Minor missions introduce a layer of strategic decision-making, providing players with an opportunity to diversify income streams and enhance the overall performance of their corporate entity.

Game Lifetime

The spirit of the game embodies an evolving and infinite gameplay concept. The player can only face defeat when the corporate capital becomes negative at the end of a game cycle (day in the game). The game's progression is characterized by gradual evolution, where benefits follow an exponential curve. This design ensures that players experience continuous challenges and opportunities for strategic decision-making, allowing the virtual corporation to thrive and expand indefinitely.

At the end of each day in the game, a summary of the day is displayed with statistics: company value, number of revenues generated that day, number of workers hired, number of new rooms built, etc.

Game Design

User interface (UI)

Respro adopts the Rowdies font, chosen for its free-to-use status. Rowdies is a simplistic yet stylized font that aligns with the game's usage. It brings a touch of fantasy without compromising a professional appearance. The interface design follows a flat design approach, prioritizing simplicity, and clarity in user interactions. This design choice aims to enhance the user experience and streamline navigation within the game environment.

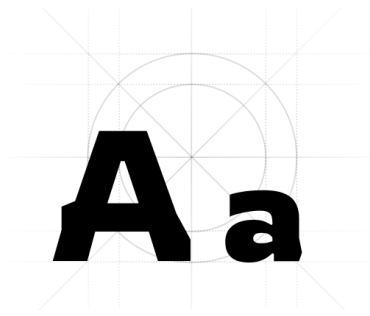


Figure 7. Rowdie font typeface.

3D Rendering

For the 3D rendering, a low-poly aesthetic is employed to reduce model complexity and enhance performance. This design decision aligns with the goal of ensuring smooth gameplay within a web browser, catering to a broader range of devices and facilitating an accessible gaming experience.

Like traditional management games, in Respro players can observe their company from an overhead perspective. This aerial view enables them to visualize the entire game board and move around it with ease. They can perform actions such as moving, zooming and tilting the camera for an immersive and intuitive experience.

Game Environment

The game takes place within a company that the player can control, represented by a skyscraper where the game board is represented by a floor. Nestled high inside the skyscraper, the game's atmosphere is enriched by a dynamic cloudy sky, changing with the hours of the day. This sky evolves throughout the day, from sunrise to daylight, sunset and night, creating a dynamic, changing atmosphere throughout the game.

Board

The company is shown as a floor within an imposing skyscraper. This floor, referred to as the board, is organized in the form of a 8x8 grid of rooms. It is on this board that the player can place and arrange the different rooms of his company, allowing him to personalize and develop his workspace.

Rooms

The rooms that the player can place freely on the board offer great diversity. Each has its own unique design, rendered in 3D in the game. For example, the open space features several shared desks, while the library has bookshelves.

In addition, each room level is associated with a variant corresponding to the room's capabilities and statistics. For example, in the open space, the number of desks increases as the room's worker capacity improves. This visual evolution reflects the progress made in the game and allows players to visualize the impact of their actions on their company's environment.

As they progress, players can notice the rooms improving and adding content. As they progress, players can see rooms improving and expanding with the addition of new elements, enriching their gaming experience within their company.

Room Walls

Each room is independent and separated by walls. These walls are dynamic and react according to the location of neighboring rooms. For example, when two rooms are placed side by side, the adjacent wall becomes a wall with a door. This adds realism and fluidity to room design, enabling players to create coherent, functional workspaces within their company.

Workers

Workers assigned to the various rooms are visible inside them, occupying positions corresponding to their functions. For example, a worker assigned to an open space will be shown seated at a desk. This feature allows players to directly observe the distribution of workers in their company.

This instant visualization also facilitates personnel management, allowing players to see briefly the number of workers assigned to each room. In this way, players can quickly assess the distribution of their human resources and make informed decisions to optimize the running of their business.

Technical Specifications

Respro game is designed as a 3D web-based game, leveraging the capabilities of ThreeJS (WEBGL) for the game engine and Svelte for the User Interface (UI). Both ThreeJS and Svelte are utilized as native frameworks to enhance development efficiency and ensure seamless integration within the web environment.

ThreeJS, a widely used JavaScript library built on top of WebGL, serves as the foundation for the game's 3D rendering and interactive capabilities. Its adaptability and performance in rendering complex 3D scenes contribute to creating a fluid gaming experience within a web browser.

On the UI front, Svelte, a component-based JavaScript framework, is employed for designing and implementing the user interface elements. Being a reactive framework, Svelte facilitates the creation of a responsive and efficient UI, ensuring a smooth and intuitive interaction for players.

3D rendering optimization

One of the major challenges in developing the game for the web lies in optimizing 3D rendering (WebGL) to ensure a smooth and responsive experience across a variety of devices. Several optimization techniques have been implemented to achieve this goal.

Firstly, one of the most straightforward yet highly effective measures is the reduction of PixelRatio to decrease texture resolution and lighten the rendering load, while still maintaining very good visual quality. This measure reduces the consumption of graphical resources without making significant compromises on quality, resulting in better overall performance of the game on browsers.

Additionally, scenes have been optimized by simplifying 3D models with a balance between visual appearance and the number of vertices. By reducing the complexity of models without compromising visual quality significantly, the rendering load has been minimized.

Another strategy is the reduction of the number of polygons in the scene. By limiting the number of polygons rendered on screen, rendering performance has been improved, especially on less powerful devices.

Finally, the extensive use of mesh instances for 3D objects sharing the same geometry, such as walls, workers, etc., has been employed. This technique optimizes resource usage by reducing the number of draws for similar objects, contributing to overall performance improvements in the game.

Optimized cross-platform rendering

Special attention has been given to optimizing rendering for a consistent experience across multiple platforms, including desktop computers and mobile devices.

Firstly, extensive work has been done to ensure compatibility with various devices, with a focus on supporting touchscreens and small-screen devices (such as small tablets). This includes adapting controls and interactions to provide a smooth and intuitive experience.

Additionally, efforts have been made to adapt responsive games for small screens, such as small tablets, by adjusting UI elements and optimizing 3D rendering for optimal visibility and playability on these devices.

Finally, porting control and 3D navigation features to touchscreens has been a priority. This involves optimizing interactions for intuitive manipulation of the view using touchscreen gestures, thereby ensuring an optimized gaming experience across all these devices.

Flow Diagrams

The coordination shows detailed flowcharts that describe all the processes and internal functions that are programmed within the application.

Technical Working

The first diagram describes the workflow required to make the application. This diagram is especially interesting since it should be used when modifying, expanding, or making subsequent versions.

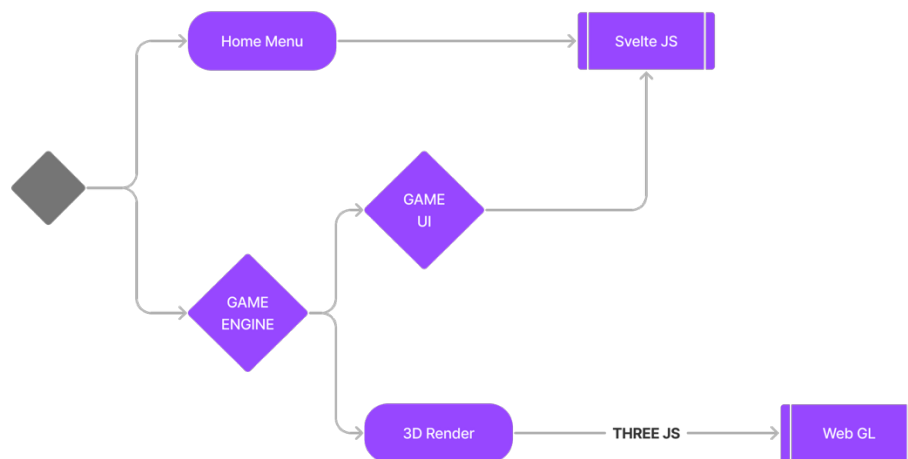


Figure 8. Technical Working Diagram.

The work diagram does not necessarily have to be the one written in the previous figure, it may be varied depending on the circumstances, platforms or changes, however these variations must be

considered very carefully since the impact on the work already developed may be very large and due to this impact it may be that the modification or extension has not been convenient or efficient to carry out.

Game Engine System Diagram

The game system machine diagram describes how the game works internally and what elements intervene for each of the actions to be performed. This diagram is especially important since it is the one that connects all the code with the elements described in the use cases.

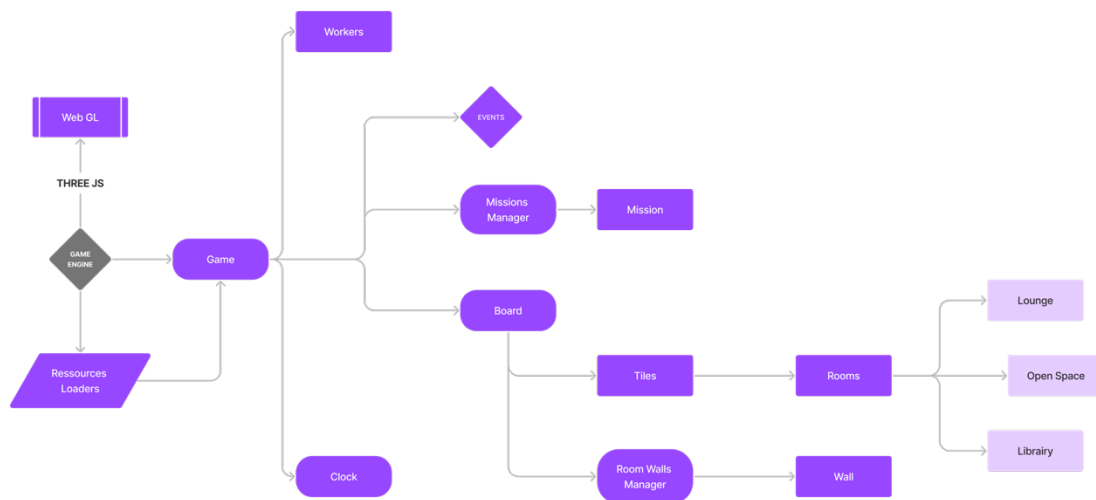


Figure 9. Game Engine System Diagram.

Obviously, any change that needs to be made to the game and that involves an expansion of elements, for example elements other than rooms or some different types of rooms or some different flows in the gameplay, should modify this diagram.

Game User Interface (UI) Diagram

The game's user interface diagram is especially important since the interface is the direct connection with the player. A modification in the interface, even if it is very tempting, should only be carried out in cases where there is a benefit to be obtained from said modification. As you can see, there are many elements, especially the buttons that lead to different playable actions and then the menus that allow you to directly manage each of the states of the game.

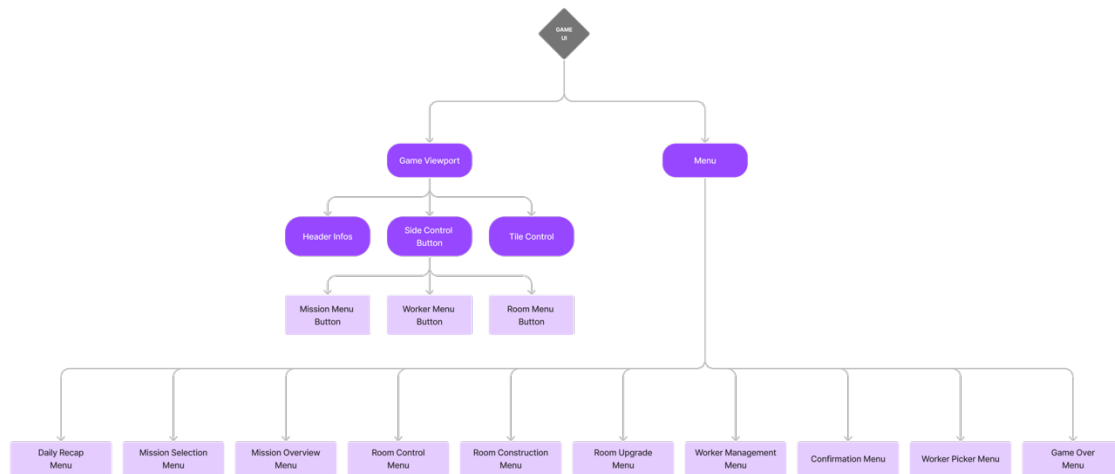


Figure 10. Game User Interface (UI) Diagram.

This diagram is highly modifiable because the different versions and extensions that may be derived from the game will probably add or remove more buttons or more menu options depending on the changes that are determined; these modifications must be made.

Game Event System Diagram

The game event diagram or system events in the game is especially relevant since it determines the flow of the gameplay. These events can be of two types: One is browser events, and the other is internal events. Browser events are those that are made directly by the user through the navigation interface, while internal events are those that happen internally in the game and determine the actions that make the game progress. Several of the events are those related to the benefit that the user takes, measured in money, apart from the temporal sequence, the spatial sequence of rooms and the events that happen to the user or in this case, the Worker.

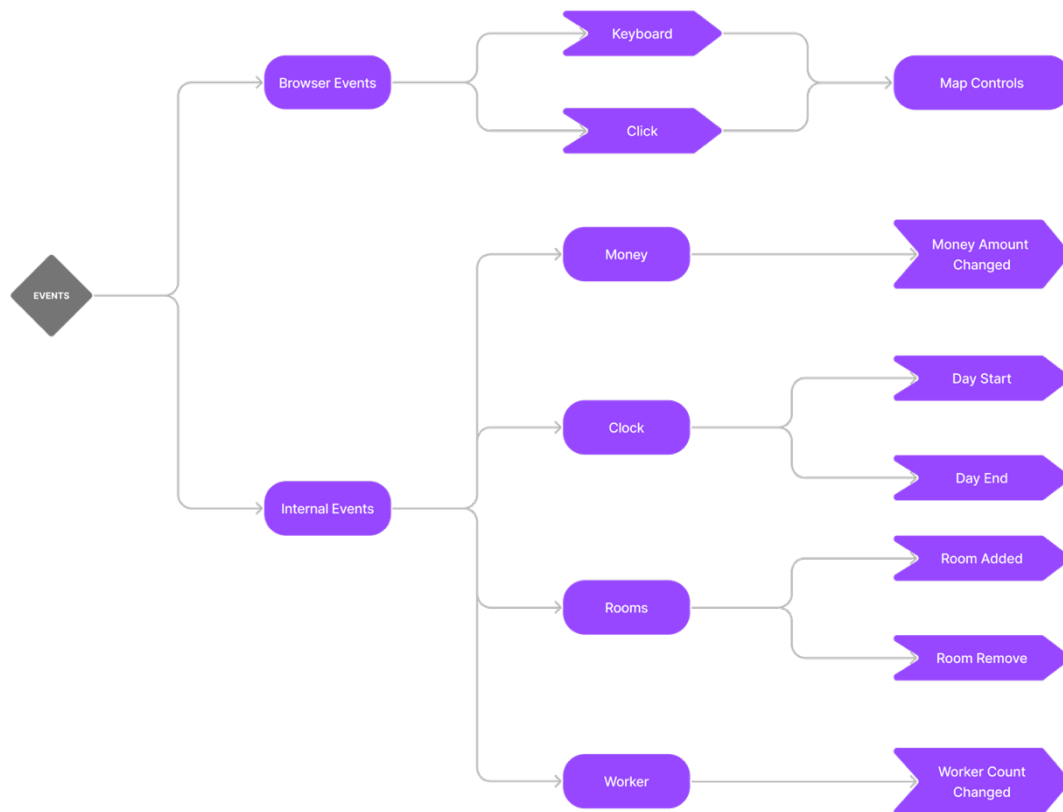


Figure 11. Game Event System Diagram.

Obviously as the game grows or new situations are added it is very likely that these events will vary, however it is recommended to maintain as much as possible the number and type of events as shown since the core control of the game is based on these and a modification of events would also imply a modification of the control core.

Conclusions

This document consolidates the technical backbone of the RESPRO e-simulation tool for WP4– R4.2 Technical Development. It translates the agreed learning journeys into a buildable, testable product and clarifies how users (players, teachers, and system administrators) will interact with the tool across the core scenarios: scenario creation, playing the game, and teachers extracting results. These use cases anchor design choices, test cases, and acceptance criteria for the release cycle.

On the engineering side, the document aligns the system architecture and technology stack with the project’s integration needs. Implementation proceeds in lockstep with the WP4 plan. The result directly supports the project’s objectives: embedding responsible leadership and well-being at work into project-management education through digital, open, and classroom-ready tools.