

Proposal

Maryam Mohammadi

Kharazmi University

Fall - 1403

Cover Letter

Dear Dylan Jadeja,

We are pleased to submit our proposal for the game development opportunity at Riot Games. With our expertise in creating engaging and innovative games, we believe we can contribute meaningfully to your vision of delivering exceptional player experiences.

We are excited about the possibility of collaborating with Riot Games and look forward to further discussions.

Thank you for your time and consideration.

Sincerely,

[Maryam Mohammadi , Ghazal Taghilu]

To Dylan Jadeja

From MQ company

Mhmd0mariam@gmail.com

Ghazaltaghilu82@gmail.com

0935406****

0930947****

Executive Summary

This proposal outlines our approach to developing the game for Riot Games. We will create an engaging and immersive gameplay experience that prioritizes smooth performance, responsive controls, and captivating visuals. Our solution will leverage advanced game development tools and technologies to ensure scalability, security, and seamless gameplay.

Company Background

History and Expertise

MQ has been delivering high-quality web and game development services for over 4 years. We specialize in creating interactive games using various programming languages, with a strong focus on Python. Our expertise extends to creating seamless user experiences with a focus on performance, functionality, and aesthetic design.

Key Personnel

Maryam Mohammadi:

- *Project Manager* – Over 3 years of experience managing and developing game projects, including expertise in leading game development teams.
- *UX/UI Designer* – Specializes in designing user-friendly and visually appealing game interfaces, enhancing player engagement and overall experience.

Ghazal Taghilu:

- [*Frontend Developer, Game Designer, etc.*]
- *Lead Frontend Developer* – Skilled in HTML, CSS, JavaScript, and responsive design, with a focus on creating dynamic user interfaces for games.
- Brings unique expertise to the team in

Project Plan

Approach

We will follow an agile development methodology to ensure flexibility and timely delivery. The project will be divided into the following phases:

1. **Requirement Analysis:** Understand and document detailed game mechanics, features, and player experience goals.
2. **Design:** Develop game concept art, wireframes, and UI/UX mockups to guide the visual and interactive design of the game.
3. **Development:** Implement core game functionalities, including character controls, enemy AI, and scoring systems, using Python and other necessary technologies.

4. **Testing:** Conduct thorough testing, including gameplay, functionality, and performance tests, to ensure a seamless player experience.
5. **Deployment:** Deploy the game to a staging environment for final approval, followed by the launch on the desired platform(s).

Timelines and Milestones

- **Requirement Analysis (December 2 - December 4):** Gather and document the game's core mechanics, including tile movement, merging rules, and scoring system. Define player experience goals and winning/losing conditions.
- **Design (December 5 - December 7):** Create wireframes and mockups for the 4x4 grid layout, tile appearance, score display, and controls. Plan the visual design, including color schemes for different tile values and overall game aesthetics.
- **Development (December 8 - December 10):** Implement core game functionalities such as tile movement, merging logic, random tile generation, and scoring system using Python. Develop keyboard controls and integrate the scoring system.
- **Testing (December 11 - December 12):** Conduct thorough testing, including gameplay, functionality, and performance tests, to ensure smooth tile movement, accurate scoring, and proper game-over conditions.
- **Deployment (December 13 - December 14):** Deploy the game to a staging environment for final approval and testing, followed by the official launch on the desired platform(s).

Technical Approach

Technologies and Tools

- **Python:** The core programming language used to implement game logic, tile movement, merging rules, and scoring system.
- **Pygame:** A Python library utilized for creating the game interface, handling tile movements, user input (keyboard controls), and rendering the 4x4 grid.
- **Tkinter:** Used for creating the graphical user interface (GUI) components, including score display and control buttons like "New Game" and "Undo."
- **Visual Studio Code:** The integrated development environment (IDE) used for writing, debugging, and managing the game's codebase.
- **Git:** For version control, managing code changes, and collaboration.

Key Features

- **Power-Ups:** Introduce power-ups that can be earned or purchased (in-app purchase optional).

Examples:

- * **Merge Booster:** Increases the chance of merging tiles.
- * **Shuffle:** Randomly shuffles the tiles on the board (used sparingly to avoid trivializing the game).
- * **Extra Life:** Grants an additional attempt after losing all lives.
- * **Tile Multiplier:** Temporarily increases the value of spawned tiles.
- **Shop:** A shop where players can spend earned coins on power-ups.
- **Difficulty Levels:** Vary the frequency of tile spawning or introduce more challenging tile combinations.
- **Daily Challenges:** Introduce daily challenges with specific goals (e.g., reach 1024 in under 30 moves) for bonus rewards.
- **Leaderboards:** Implement leaderboards to foster competition and replayability.
- **Sound Effects:** Add satisfying sound effects for tile merges, movements, and power-up activations.
- **Visual Enhancements:** Improve the visual appeal of the game with polished tile designs and animations.
- **Themes:** Offer different themes (colors, tile styles) to customize the game's look.

Cost Proposal

Description	Cost
Requirement Analysis	150,000
Game Design	1,000,000
Development	1,700,000
Testing and QA	550,000
Deployment	250,000
Post-Launch Support	450,000
Total	4,100,000

Previous Work

Similar Projects

- **Space Invaders Clone:** A 2D shooting game where the player controls a spaceship, aiming to shoot down waves of enemies. The project involved sprite management, collision detection, and sound effects integration.
- **Maze Runner:** A Python-based maze exploration game where players navigate a series of levels filled with challenges, solving puzzles while avoiding obstacles.
- **Pong Remake:** A classic two-player game recreated using Python, focusing on smooth gameplay mechanics, basic physics for the ball movement, and an intuitive control system for the paddles.

References

References

- **Pong Remake**
 - Email: pongremakesupport@gmail.com
 - Phone: 0935*****
 - Project: Developed a classic Pong remake in Python, incorporating smooth paddle controls, ball physics, scoring system, and a simple graphical interface..
- **Space Invaders Clone**
 - Email: spaceinvaderssupport@gmail.com
 - Phone: 0935*****
 - Project: Built a 2D shooting game in Python featuring sprite management, enemy waves, and sound effects.

Terms and Conditions

- The project will commence upon signing of the contract and receipt of an initial payment of 20% of the total project cost.
- The remaining payment will be made in installments based on the completion of specific project milestones (e.g., Design Phase, Development Phase, Testing Phase).

- Any additional features, changes in scope, or modifications to the original project plan will be subject to a change order and may incur additional costs.
- The project timeline is subject to adjustments based on timely feedback and approval at each stage.
- Intellectual property rights for the game will be transferred upon full payment.
- Bug fixes and minor adjustments will be included in the maintenance phase, which will last 30 days after deployment.

Contact Information

For any queries or further information, please contact:

- **Name:** Maryam Mohammadi

- **Title:** *Project Manager*

- **Email:** Mhmd0mariam@gmail.com

- **Phone:** 0935406****

- **Name:** Ghazal Taghilu

- **Title:** Game Designer

- **Email:** Ghazaltaghilu82@gmail.com

- **Phone:** 0930947****