Criteria A

[Scenario 2](#_Toc184832796)

[Solution 2](#_Toc184832797)

[Rationale 2](#_Toc184832798)

[Interaction With Client/Advisor 3](#_Toc184832799)

[Success Criteria 4](#_Toc184832800)

**//500 words**

# Scenario

**//identify client, customer, business and tech context**

My uncle's (client) company, which has successfully developed engaging single-player (customer) games, now aims to venture into multiplayer games. The primary motivations for this shift include market expansion by attracting a wider audience, enhanced user engagement through social interaction, community building around the company, and revenue growth. By offering multiplayer games, the company can diversify its product range, increase user retention, and build a loyal player community, ultimately driving higher profitability.

# Solution

**//explain functional, technical, devlpt model**

We will develop two-player turn-based games, specifically Tic Tac Toe and Checkers, delivered as a client Java application to the players. This application will connect to a centralized database managed by my uncle's company. Players will play together on the same application instance in the same Java virtual machine. We ll use AGILE/SPIRAL/WATERFALL development model.

# Rationale

**//explain business, functional, technical, devlpt model , costs, timing rationale that is important for the solution**

**Business Rationale**: Developing two-player games like Tic Tac Toe and Checkers is a quick-to-market solution that creates dynamic experiences and prepares our platform for future more-than-2 player games. This approach allows us to evaluate market insertion cost-effectively before making larger investments. By starting with simpler games, we can build a foundation for more complex multiplayer games, ensuring a gradual and sustainable growth strategy.

**Technical Rationale**: Using Java as the client technology ensures operability on all devices, speeds up development with its extensive libraries, and benefits from strong community support. A central relational database like MariaDB maintains data integrity, scales player growth, supports complex queries, and simplifies maintenance, making it ideal for managing multiplayer games.

**Methodology rational**: The Agile methodology is chosen for its flexibility and adaptability. It supports iterative development and continuous feedback, making it easier to adapt to changes and improve the games incrementally. Regular functional increments enable early stakeholder feedback, ensuring the games meet expectations. Agile also promotes regular communication and collaboration, crucial for aligning everyone and addressing issues promptly.

**Or**

The Spiral methodology is chosen for its focus on risk management and iterative refinement. It allows for continuous risk analysis and mitigation, making it easier to address issues early. Iterative refinement ensures regular prototyping and validation, meeting user needs and enabling incremental improvements. This approach combines flexibility with structure, promoting regular reviews and adjustments based on feedback.

**Or**

The Waterfall methodology is chosen for its clear and stable requirements. Its sequential progression ensures thorough documentation and adherence to initial specifications, making planning and execution easier. Predictability and control over timelines provide a clear roadmap, ensuring the project stays on schedule and all aspects are carefully managed.

//define advisor

In this context, my uncle’s CTO, who has extensive experience in game development and database management, will act as an advisor, providing valuable insights and guidance throughout the project. This ensures that the project leverages the best practices and industry standards, leading to a successful implementation of the multiplayer games.

*// Give online references for each paragraph or annexes…*

# Interaction With Client/Advisor

**//demonstrate how you interact for requirement, dev, review and deploy**

**//if you can describe an example of discussion about a critical point of your case(functional, or cost or technical)**

**//For Agile**

We hold regular meetings, including sprint planning, and sprint reviews, to ensure continuous feedback and iterative development. This approach allows us to adapt quickly to changes and ensure the games meet client expectations through frequent communication and collaboration. We defined stories and requirements using conceptual modelisation through diagrams and tables, interviews and previous solution analysis.

**//For Spiral**

We conduct initial planning and risk assessment sessions, followed by iterative prototyping and validation meetings. This method focuses on continuous risk management and iterative refinement, ensuring early identification and mitigation of potential issues. We defined use cases and requirements using conceptual modelisation through diagrams and tables, interviews and previous solution analysis.

**//For Waterfall**

We start with comprehensive requirement gathering and design reviews, followed by regular progress check-ins and a final review meeting. This linear approach ensures thorough documentation and adherence to initial specifications, providing a clear roadmap and predictable project timeline. We defined stories and requirements using conceptual modelisation through diagrams and tables, interviews and previous solution analysis.

*// Give annexes showing that (reunion resume, call record…), any evidence is fine*

# Success Criteria

//4 to 10 maximum

1. Implement two games: Tic Tac Toe and Checkers.
2. Players are playing in the same java application (no remote)
3. Allow players to choose which game to play.
4. Manage player name, including creation, selection, correction, and deletion.
5. Display game boards, player names, and turns clearly.
6. Update and display scores and rankings after and before each game.
7. Penalize players for canceling a mid-play game.