Creating a bluff game with online multiplayer functionality and Tezos blockchain integration is a complex task, but I can guide you through the high-level steps to get you started. Keep in mind that this is a broad overview, and the actual implementation may require more in-depth knowledge and skills in web development, blockchain integration, and networking. Let's break down the process into several steps:

Define Game Rules and Logic:
Before diving into development, define the rules and logic of your bluff game. Decide how many players can participate, the card distribution process, the winning conditions, and the bluffing mechanics.

Frontend Development:
Start by creating the frontend of your game using HTML, CSS, and JavaScript.

Design the game interface, including the game board, cards, and chat box. Make sure the UI is user-friendly and responsive.
Implement the card distribution animation and game animations for smooth user experience.

Backend Development:
Set up a backend server to handle game logic, card distribution, player actions, and chat messages.
You can use Node.js or any other backend technology you are comfortable with.

Online Multiplayer Functionality:
Implement the multiplayer functionality to allow players to join the game and play together.

Use WebSockets or a similar technology for real-time communication between players and the server.
Handle player connections, disconnections, and game synchronization.

Tezos Blockchain Integration:
To integrate Tezos blockchain, you'll need a Tezos wallet for each player. Players will need to sign transactions, so consider using a library like Taquito (JavaScript library for Tezos) to interact with the Tezos blockchain.
Define smart contracts that handle the game state, player actions, and verify the validity of moves.
Implement the communication between your backend and the Tezos blockchain to update the game state and manage transactions.

Chat Functionality:
Develop the chat feature to enable players to communicate with each other during the game.

Implement message broadcasting and display messages in the chat box.

User Authentication:
Implement a user authentication system to identify players and ensure the security of their actions.
Consider using OAuth or JWT (JSON Web Tokens) for secure authentication.

Testing and Bug Fixing:
Thoroughly test your game to identify and fix any bugs or issues.

Test the game with multiple players simultaneously to ensure smooth gameplay.

Deployment:
Deploy your frontend and backend to a hosting provider.

Ensure that your server can handle the expected number of players.

Maintenance and Updates:

Regularly maintain and update your game to improve its functionality and security.

Listen to user feedback and make necessary changes accordingly.

Keep in mind that this is a complex project that requires expertise in frontend and backend development, blockchain integration, and networking. Be prepared for challenges and don't hesitate to seek help from online resources, communities, and forums as you work on your bluff game project. Good luck!