

# Research Plan

## Belote Online Game – Dimitar Atanasov

### 1. Introduction

For this semester, 'Complex Software Systems', I have decided to undertake an individual project to develop an online card game. When brainstorming how to approach this goal, multiple ways of bringing it to life came to mind. The first task I thought about was how I wanted to create the visual aspects of the game, the ones the user will interact with. There are many technologies I could apply, and I am fairly confident in my abilities to complete that. Then my mind came to the real difficult system of my undertaking – multiplayer synchronisation. Being completely unfamiliar with technologies and strategies to implement this, I have decided to make this my main research topic. I intend to gain a very deep understanding of multiplayer syncing strategies, the problems that accompany them and relevant solutions. I hope this knowledge gained will also contribute to my career ambitions of eventually breaking through in game development.

### 2. Problem / Opportunity

Synchronising game states between multiple clients and a server is a fundamental challenge for most spheres of modern day gaming. Unlike singleplayer games, online multiplayer games must ensure that all players interact with the same state, their actions affect each other's playing experiences, all while maintaining low latency and preventing exploits and cheating. While my project is a smaller in scale turn-based card game and does not require advanced anti-cheat systems, the fundamental knowledge I gain from this will be applicable everywhere. Without proper multiplayer synchronisation, my card game will be unplayable. This problem therefore provides me with an opportunity to research a topic that will allow me to build extensibly larger games and have a large RoI for my future career.

### 3. Research Questions

#### Main Question:

What strategies, technologies, and design patterns are most suitable for implementing reliable multiplayer synchronisation in an online turn-based card game?

To answer this main research question, I will split it up into smaller sub-questions:

| Sub-questions   | Strategies                         | Methods   |
|---|------------------------------------|---|
| 1. What are the common synchronisation models used in multiplayer games (e.g., client-server, peer-to-peer, authoritative server)?    | Library, Showroom, Workshop        | Literature study, Design pattern research, IT architecture sketching, Peer review   |
| 2. How do different communication technologies (WebSockets, gRPC, HTTP polling) compare in terms of suitability for turn-based games? | Library, Lab, Workshop             | Best good and bad practices, Available product analysis, Benchmark test, Multi-criteria decision making                   |
| 3. What role do databases and caching systems (e.g., Redis) play in maintaining game state consistency and persistence?               | Library Lab, Showroom              | Literature study, Community research, Component test, Guideline conformity analysis                                       |
| 4. What are the main failure scenarios (disconnects, latency spikes, cheating attempts), and how can they be mitigated?               | Library, Field, Lab, Workshop      | Best good and bad practices, Stakeholder analysis, Computer simulation, Root cause analysis                               |
| 5. How do scalability considerations affect multiplayer synchronisation design?   | Library, Field, Lab, Workshop      | Competitive analysis, Domain modelling, System test, Business case exploration  |
| 6. How can microservices and Kubernetes be used to deploy and scale an online turn-based card game?                                   | Library, Field, Workshop, Showroom | Design pattern research, Available product analysis, Problem analysis, IT architecture sketching, Static program analysis |