**Chunked Design**

Chunk the design into discrete sub-systems using an Affinity Spreadsheet

**Regular Gameplay**

|  |  |
| --- | --- |
| **Game Subsystem** | **Activities** |
| **Turn Management System** | Allocate player piece to player (random assignment) |
|  | Turn Assignment |
|  | Shuffle Rock Cards |
|  | Prepare Deck |
| **Game Elements** | Invoke game element |
|  | Verify if the answer is correct |
| **Turn Elements** | Start Turn |
|  | End Turn |
| **Piece Movement** | Dice Outcome |
| **Game Termination** | End Game |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Game** | | | | |  |
| **Operator** | **Randomization System** | **Quiz Management System** | **Scenario System** | **Trading System** | **Turn Management System** | **Game Management System** | **System State** |
|  | Provide Random Events | Quiz Cards given out | Give out Scenario | Trading Counter | Manage Play according to game rules | Rock Cards tracking |  |
|  |  | Read Quiz Questions | Dissolve | Provide rules to trade | Track turns | Scenario Cards tracking |  |
|  |  | Verify Quiz Answers | Cleave | Manage the exchange rate | Assign player turn order | Quiz Cards tracking |  |
|  |  | Quiz Cards Answered Correctly | Melt |  | Pass dice to next player | Dice rolls |  |
|  |  | Quiz Cards Answered Wrongly | Scratch |  | Roll Dice | Shuffle Cards |  |
|  |  |  | Steal |  |  | Card Dealing |  |
|  |  |  | Crush |  |  | Discard card |  |
|  |  |  |  |  |  | Trivia Information checking |  |
|  |  |  |  |  |  | Player locations tracking |  |
|  |  |  |  |  |  | Impose Scenario |  |
|  |  |  |  |  |  | Pick up card |  |

**Modularity**

1. **Design the tiles**
2. **Design the Quiz**
3. **Design the Rocks**

**Your project design must include at least three interconnected modules:**

**Design Rules:**

E.g.

* System/360 instruction set
* Hardwired processor microinstruction
* Microcode to convert System/360 instructions to processor instructions (except for largest processor)
* Compilers written in machine language (based only on System/360 instruction set)
* IOCS implemented in special purpose processors (“channels”)
* Only four standard I/O commands for processors

**Interfaces (Current):**

**Interfaces (Future):**

* Memory-CPU coupling same for all 5 processors
* Any I/O device sees identical channel
* I/O control system common across family