

# Analysis of Mechanics: League of Legends

Dávid Čislák (xcislad00)

- **Title:** League of Legends
- **Released:** 2009
- **Author:** Riot Games
- **Primary Genre:** MOBA

- **Secondary Genre:** RTS, Action RPG, Tower Defense, Roguelike
- **Style:** Stylized High-Fantasy, Isometric

## Reflection of Genres in Gameplay

League of Legends is one of the games defining the MOBA genre, a game genre descended from the RTS modification *Defense of the Ancients*. The primary genre is reflected in the core objective of the game: two opposing teams of five players interact within a static, three-laned map *Summoner's Rift* with the objective of destroying the opposing structure *Nexus*. However, unlike traditional RTS games where a player controls armies, the gameplay is focused on the control of a single unit—the *Champion*.

The secondary genres (Action RPG, RTS, Roguelike, and Tower Defense) build upon the primary genre and enhance it. The game takes most of its control scheme from the RTS genre (isometric camera, right-click movement, fog of war, and strategic map awareness). The Action RPG genre dictates the character progression. Each match begins with players at an equal footing (level 1), representing session-based progression similar to a Roguelike run. The players cannot attack the Nexus right from the beginning, they must first destroy the enemy defending structures (towers) while protecting their own.

To effectively deal damage to these structures and avoid their defensive mechanisms, the player has to lead allied non-player entities (minions) to the enemy tower so they absorb the tower's attacks. The allied and enemy minions spawn at the same time near the nexus and walk in a straight line. They spawn in groups of six to seven based on the time of the game. The minions on the path to a structure do not ignore enemy units or champions. This creates a push and pull flow where the player helps the minions push through enemy units, resembling Tower Defense mechanics.

These minions and other non-player units are the backbone of the in-game economy. With each player starting at the lowest level with zero equipment, they are required to engage in the slaying (farming) of non-player entities to accumulate gold and experience. This allows players to purchase items and upgrade abilities, reflecting a condensed RPG power curve that occurs over 30 to 40 minutes. Character leveling increases Champion stats like health and damage, but its main purpose is to learn and level up abilities unique to the champion. Most champions have predetermined roles like mage, fighter, or assassin, but players can shift the Champion's stats and role based on purchased items.

The Action component is realized through fast combat and skirmishes with enemy players, requiring precise aim for skill-shots and split-second reaction times to dodge enemy aggression, moving the game away from pure strategy and emphasizing mechanical skill.

## Interaction of Primary and Secondary Genres

The genres within League of Legends work together to create a gameplay loop that balances strategic planning and me-

chanical execution. This is most visible in the interaction between the *Macro* and *Micro* layers of the game.

The RTS and Tower Defense establish the *Macro* layer. The Tower Defense mechanics, specifically the spawning and pathing of minion waves, dictate the pacing. These waves force players to be present in specific locations (lanes) to collect resources. However, the RTS elements, such as the Fog of War, introduce danger. Players must predict enemy movements, deciding when to follow the Tower Defense structure (staying in lane) or when to break it to ambush enemies or capture objectives.

The interaction between Action and RPG elements defines the *Micro* layer. While the RPG genre provides the mathematical foundation (stats, cooldowns), the outcome of an engagement is rarely decided by numbers alone. The Action component disrupts the traditional RPG "stat-check", a player with lower statistical values can overcome a stronger opponent through superior mechanical execution. This creates a high skill ceiling where RPG knowledge must be paired with good reflexes and mechanical skill.

Finally, the Roguelike structure serves as the spine for the game's economy. Because every match resets progress, the "condensed power curve" forces an urgency absent in traditional RPGs. Players must optimize their farming efficiency within strict time constraints. A small advantage gained early translates into an RPG statistical advantage, allowing the team to deny enemy resources, capture objectives and eventually destroy the Nexus.

This genre combination ensures the game remains compelling regardless of the match state. For the disadvantaged team, the Action component offers a chance to recover ("comeback"), the RTS layer allows for asymmetric strategies such as "split-pushing" or "backdooring", where a single player threatens the enemy Nexus. For the winning team, the RPG progression enables a "power fantasy," allowing a player to "snowball" and defeat multiple enemies alone and validate the structure of the condensed power curve.

## Style and Visual Hierarchy

The visual style is a functional choice designed to support the chaos of gameplay. Riot Games uses a "stylized realism" or "hand-painted" art style. This is crucial for visual clarity. In a team fight, up to ten players may cast multiple spells at once, creating visual noise.

The champions use expressive animations to adapt to the isometric view. During some animations, the character model stretches unnaturally to create an illusion of speed. By using exaggerated silhouettes of the characters, bright colors, and distinct spell effects, the style allows players to instantly recognize threats. Important abilities are visually louder than auto-attacks. The fantasy setting allows distinct thematic elements (a steampunk sniper fighting a dragon) to coexist without breaking immersion. This stylized approach also ensures accessibility on lower-end hardware and ages better than games that seek graphical fidelity.