



Robot Factory

Mobile game for learning and practicing computer science concepts

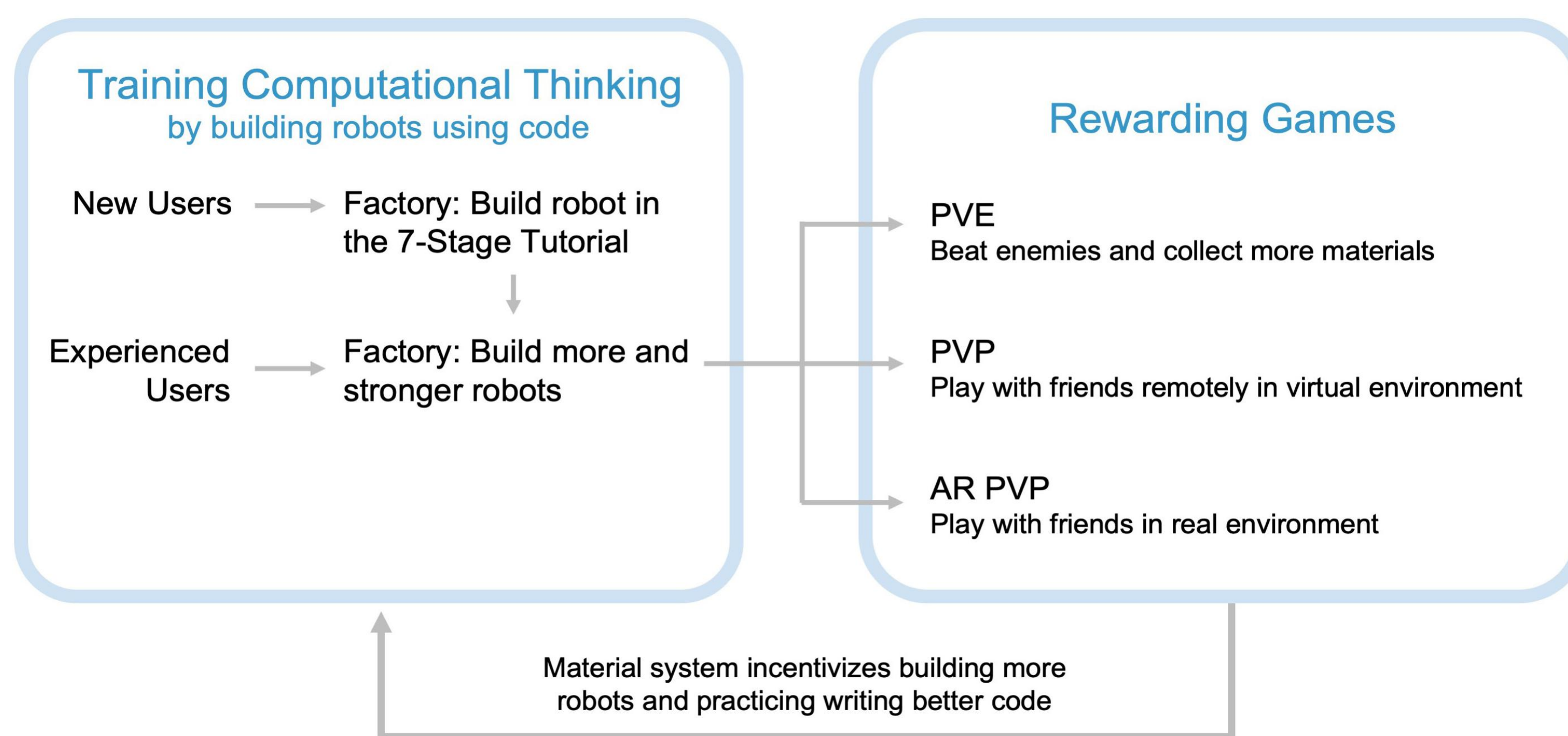
Problem
Over one million computer science jobs will remain unfilled in the United States in 2020. No formal curriculum or education plan exists to close these gaps in this expanding industry.

The combination of no formal CS education in middle schools and a lack of qualified CS teachers in K-12 schools presents an opportunity. How do we fulfill society's need for human resources in computer science while building interest and competency in this field?

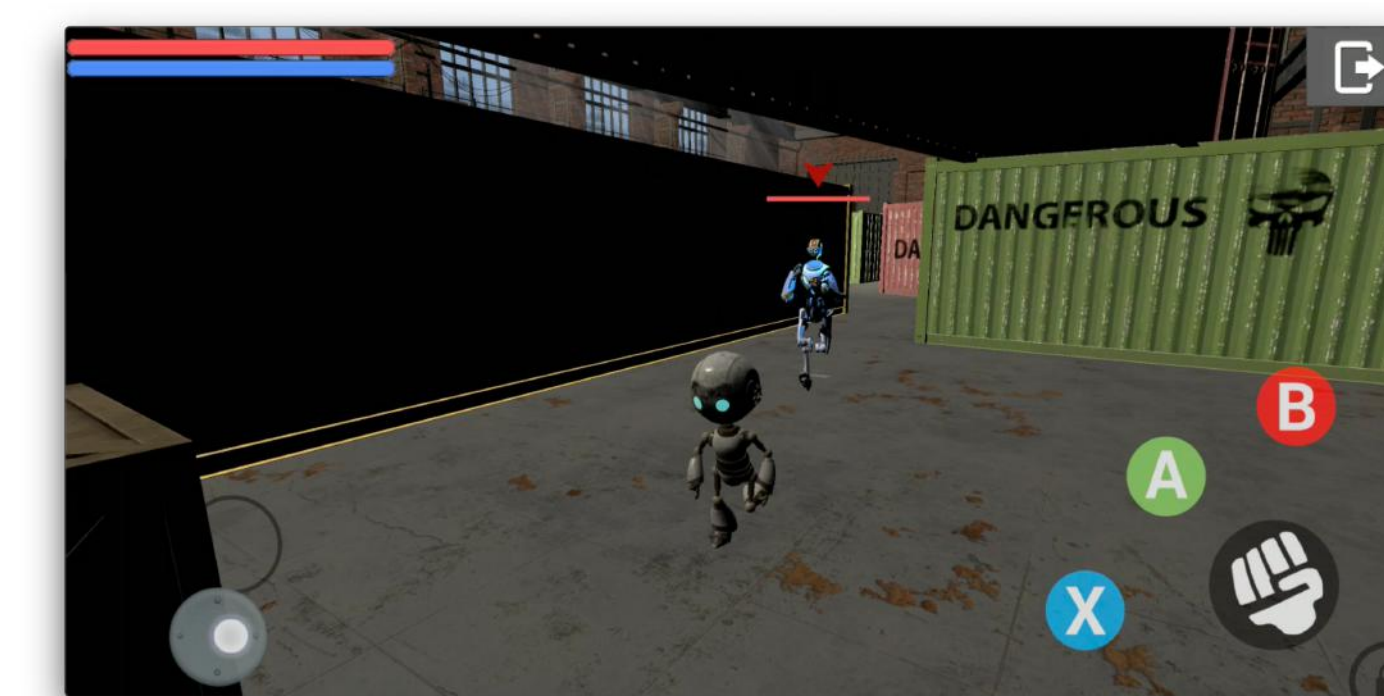
Solution
Robot Factory targets a variety of player archetypes, including the collector, explorer, socializer, and achiever. The three-part narrative arc combines unique gameplay objectives to set up, build tension, and reward players.

Programming and computational thinking happens in the robot factory, where players can build various robots with different capabilities and strengths while gaining a sense of ownership.

In the PVE (player vs. environment) scene, players can control the robots they build, collect materials, and use them in more sophisticated programming challenges. Finally, players meet in fully simulated or augmented reality environments to socialize or compete.



Game Mechanics



Player vs Environment (material collection)



Player vs Player Virtual Space



Factory part: writing code



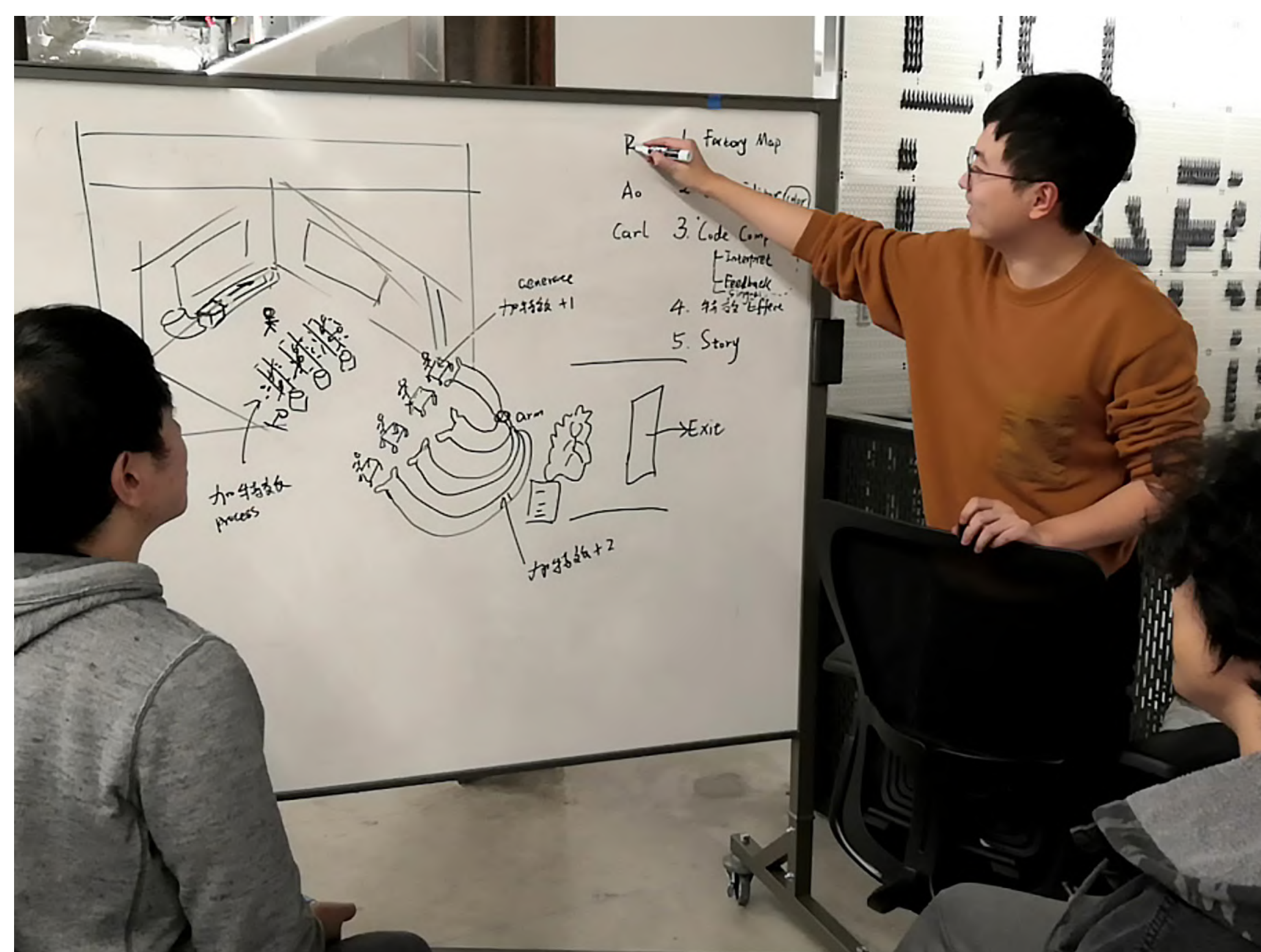
Factory part: animation



Augmented Reality Player vs Player



Playtesting social-competitive challenges



Designing factory stages

Process/Approach

Middle school students are highly adaptable learners with sufficient math proficiency, capable of understanding a variety of programming concepts. To target middle schoolers, we chose mobile games, as we found nearly 75% of households have a person who identifies as a gamer, and 45% of middle schoolers have personal mobile phones. Even more have regular access to mobile devices. Kids playing our game can familiarize themselves with programming concepts while having a fun objective they can work towards.

The core game mechanic utilizes computational thinking and logic, delivered in a concise and entertaining programming interface. To make the act of programming more fun, an economy of rewards entices the players to improve their code and advance their robot building skills.

In design evaluations, we examined the game's usability, learning deltas, fun factor, and overall sentiment of the playtesters. We pre-assessed computer science knowledge and experience with a qualifier survey and provided tutorials for each play session. After the playtests, we evaluated learning and sentiment, focusing on computational knowledge and skills gained.