Executive Summary

An Executive Summary Submitted to the

Faculty of the School of Engineering and Applied Science

University of Virginia - Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Alan Li

Spring, 2023

On my honor as a University student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments.

ADVISORS

Joshua Earle, Department of Engineering and Society

Briana Morrison, Department of Computer Science

Introduction

This portfolio has two sections: one about my STS research and the other about my technical project. The STS research delves into factors or characteristics of gacha games that have led to their growth in the United States and China while also analyzing them through a lens of technological citizenship and actor-network theory STS frameworks. The second part of my portfolio goes over my technical project which involved creating an online coding playground utilizing a new browser-based technology called WebAssembly. This executive summary gives a glimpse into what the two papers will delve deeper into.

STS Paper

Virtual gambling games, known as gacha games, have seen unprecedented growth in the United States (U.S.) due to their progressively addicting designs. These games are characterized by their unique in-game currencies which are used to roll - like trying to win the lottery - for in-game items or characters. Many gacha games have seen significant successes globally and the paper will dive into a few of them as well as the history behind how the genre started and the road it has taken to take up increasingly large market shares of the mobile gaming industry.

With gacha being a relatively new experience for many new players of the genre, their well-being needs to be taken into consideration when the government will offer little help in regulation and companies seek to maximize their own profits in the growing market. In the paper, a literature analysis will be completed on several sources that focus on the growth of gacha games in the U.S. and China and the differences in policies or perspectives that gacha games face in the two countries. The methods will be explained in the research and methods portion of the paper with research findings compiled after and finally an analysis of how the two countries compare as well as potential future steps that can be taken to protect the gacha gaming community as its popularity finds new players.

Research Question: What factors have led to the difference in the growth of gacha games in the United States vs China?

With the gacha game market growing larger in the U.S. and the lack of any regulation set by the U.S. government, it will be critical to analyze if this rapid growth has led to any potential hidden consequences. If the rates of in-game spending are growing much faster than those found in veteran communities, some type of intervention might be required, or further research as to why this is happening will also be needed.

Many aspects affect how fast gacha games can grow in the U.S. and a lack of knowledge of this particular community can exasperate these aspects. Various perspectives of the genre have been studied including but not limited to its addictive nature/properties, smartphone usage, growth of mobile gaming, and policies aimed towards curbing the time and money spent. These perspectives will be analyzed in the paper to better understand how large a problem gacha game addiction could prove to be in the U.S.

Technical Project

An Arlington, VA-based cloud computing company required a web coding playground for one of its new services enabling users to input code and evaluate its validity and output. However, some customers required a local application for their confidential data. I was tasked with designing and building a React.js web app to serve as the frontend user experience but required a backend code evaluator without sending data to and from the customer and developers. The solution was to use WebAssembly to compile backend Rust code to an assembly and JavaScript package which was then used in the React.js web app to evaluate code locally.

With these two components connected, the app was fully functioning locally, without a need for any network calls. The result was a much faster application as initial evaluation calls were 50 times faster and subsequent calls 100 times faster than a similar playground using API calls. While the application was complete, many user interface designs were not fully implemented, and an online hosted version required for internal use had not yet been completed.

Table of Contents