Chengzhe Li

(+86) 18627971711 chengzheli@umass.edu http://cunzhe.life https://linkedin.com/in/cunzhe

EDUCATION

UNIVERSITY OF MASSACHUSETTS AMHERST **2020-2021**
M.S. Computer Science Candidate, CICS *GPA:4.0*

UNIVERSITY OF MASSACHUSETTS AMHERST **2016-2019**
B.S. Computer Science, College of Information and Computer Science *GPA:3.6*

- **Awards & Activity:** Bay State Master, Course contributor of Computer Networks, ESL Activity Assistant

PROFESSIONAL EXPERIENCE

TENCENT **SHENZHEN, CHINA**
Software Engineer Intern *May 2020 - July 2020*

- Worked as a part of IEG(Interactive Entertaining Group)-CTG(Core Technology Group), our project is a distributed framework for improving the efficiency of Unreal Engine. The framework is developed in .NET and can be used in UBT(Unreal Build Tools), Lightmass, and ShaderCompiling
- Implemented a new hybrid distributing function and a test framework. Shortened the CPU sync time by **70%**
- Conceptualized and implemented the testing framework with 6 distributing methods to evaluate the improvements with configurable parameters, reported the research finding
- Tested, found and fixed concurrency bugs in the framework
- Worked with CI to make the project pipelined and efficient. Used data visualization tools to analyze logs
- Learned how to get into a large project, work in an **Agile** team to deliver the product quickly and correctly, finding and fixing problems, develop and test with tools, adding new functions to an existing framework etc.

PROJECTS

Pygmy

- Pygmy is a simulated online bookstore using micro services. We built frontend server, order server, and catalog server as three separate RESTful micro services by Flask, and a simple command-line user interface to interact with it. Worked in a team with two
- Used **Sqlite3** as the lightweight database and storing information on catalog server. The frontend server supports three operations, search(category), lookup(item), and buy(item). The first two query the catalog server, the third one calls the order server, then the order server will update the catalog server and return
- Besides the basic functionalities, we implemented support for replication and synchronization on backend servers, cache on frontend servers, load balancing, and crash fault tolerance and recovery. We also used **Docker** to containerize our microservers so it’s easier to deploy

Bazaar

- P2P networks built in Python, used pyro4 as RPC library, nodes were able to raise lookup call to buy items using Gossip protocol. The message would be passed by RPC function. The structure of the network was configurable. Multiple transactions can be processed at the same time. Worked in a team with two

CookingPapa

- Game developed used Unity. The idea is the player needs to fight with the monsterized animal or vegetable to get the food material. Some other material will be hidden on the map or looted by some other ways. After gathering all materials, there will be a real-life simulated cooking stage for the player to play and learn
- Chose the assets used, deigned the game logic, and implemented the gameplay. Set up the scenes, UI, etc..
- Added videos, BGMs, and sound effects to make the game more enjoyable
- Added clear instructions, particle systems, and physical to make the game harder but more interesting
- The playable game and demo video are available on my Github

SKILLS

- Proficient in C#, C/C++, and Python, familiar about Java, and always willing to learn new languages
- Knowledge and experience of implementing and using AI algorithms and ML models and concepts
- Understanding of distributed systems, complicated operating systems, and secure decentralized systems
- Programming experience with RESTful, RPC, multithreading distributed system, familiar with docker
- Developing experience in Unity and UE, knowledge about game programming and underneath
- A good knowledge base of Computer Graphics/Vision, Linear Algebra, and software engineering
- Passion on Technology, Music, Games, and new areas. I’m always hungry to learn, and good at it!