# Runfei **Wu**

## Msc student in Computer Science, Københavns Universitet

linkedin.com/in/runfei-wu github.com/JasonWurunfei
+45 52 63 73 60 jasonwurunfei@gmail.com
Øresundsvej 9, 2300, Copenhagen, Denmark


I am a highly motivated and adaptable team player seeking to thrive in IT development, striving for excellence, eager to embrace challenges, and committed to continuous learning and career advancement.

## Education

2022 - now **Master of Science in Computer Science, University of Copenhagen, Denmark**
**Courses :** Advanced Programming(10/12), Advanced Algorithms and Data Structures(12/12), Computational Geometry(12/12), Computational Methods in Simulation(12/12), Signal and Image Processing(10/12), Proactive Computer Security(12/12), Machine Learning A, Advanced Computer Systems

2018 - 2022 **Bachelor of Science in Information Processing Science, University of Oulu, Finland**
**Bachelor of Engineering in Software Engineering, Nanjing Institute of Technology, China**
**Courses :** Discrete Structures, Software Architectures, Data Modeling and Design, Software Development, Maintenance and Operations, Databases

## Work Experience

Now
August 2023

**Game programmer, The National Academy of Digital, Interactive Entertainment (DADIU), Copenhagen, Denmark**

- Developing a Movement-based adventure game where you sail through a desert of tormented souls in order to mend an unbalanced world with magical thread. The game is developed using Unity. I am in charge of developing core game mechanics and work as a bridge between the developers and game designers. Implementing camera positioning and Fmod Audio integration.
  Studio Linkedin

Unity | Plastic SCM | C# | Scrum

August 2021
July 2021

**Software Engineer Intern, Nanjing TDH Technology Co Ltd, Nanjing, China**

- Developing wrapper interface around the company's old testing tool (mercury quality center) to provide a more modern interface for testers to write and perform test cases much faster. Adding additional features such as generating real-time reports and data export to various formats. This wrapper interface saves the company plenty of time as the reporting and data export were partially done manually before
- Writing evaluation documents of new testing platforms for the company. Developing data migration tools to move the company's existing data to a new testing platform (MeterSphere).

Python | SQL Serve | Vue.js | Restful API

## Projets

**Trippy Travel Chatbot** Dec, 2021

github.com/JasonWurunfei/trippy-rasa
github.com/JasonWurunfei/trippy-web
github.com/JasonWurunfei/trippy-db-api

- A chatbot for an imaginary long-standing travel company which plans to deploy a chatbot to assist customers with booking travel plans, provide information on popular packages, and handle specific grievances. The chatbot's key functions include offering details on popular destinations, aiding in booking, and addressing flight, tour guide, and hotel issues.
- The system was developed in three parts, the chatbot itself uses Rasa, a conversational AI software for building text and voice-based assistants, a backend server implementing Restful API, and frontend web pages. The development process adopted continuous integration (CI) and continuous development (CD) practises by containerized the services and using Jenkins together with Github to build the deployment pipeline.

Python | HTML/CSS/JS | Docker | Jenkins | CI/CD | Rasa | NLP | Azure

### Marble - Language Learning App — Jun, 2021

github.com/JasonWurunfei/Marble-Web
github.com/JasonWurunfei/Marble-API

- Marble is a software product designed to assist proactive language learners in gradually accumulating bilingual or multilingual proficiency through daily interactions. It is multimedia-based, allowing users to record personal experiences, observations, and events through videos, photos, and more. These records can then be linked to words or sentences in the target language.
- The project was implemented with the Spring boot framework. The services were containerized. The tests and deployment were automated by Jenkins. The web app has a responsive design for both mobile and PC platforms.

Java | Spring boot | Multimedia data | Docker | Jenkins | CI/CD | Responsive UI

### Handcraft Mult-thread HTTP Server — Oct 18, 2020

github.com/JasonWurunfei/programming3-ChatSys

- Created TCP and HTTP chat server classes with message protocol support
- Implemented CLI client for server connectivity.
- Supported three database systems (text, SQLite, MySQL)
- Developed secure server and client using SSL/TLS
- Practiced Test-Driven Development (TDD) throughout the development process
- Documented methods with Javadoc for clarity.

Java | Multi-threading | TCP | HTTP | SQLite | MySQL | SSL/TLS | TDD | Javadoc

### Dynamic Hyper-elastic Material Simulation — May, 2023

github.com/JasonWurunfei/dynamic_hyper-elastic_materials_simulation

- 2D simulator to simulate dynamic hyper-elastic materials. The setup is a cantilever beam with the left boundary fixed to a rigid wall and the right boundary with some traction load. Meanwhile, the Gravity is applied. The simulation is to calculate the deformation of the beam caused by the traction and body force at every time step.
- The simulation uses the Finite Volume Method with median dual-centred vertex control volume to discretize the domain and uses the Finite Difference Method, the first-order semi-implicit Euler time integration method to solve the time derivative.

Python | Numpy | Computational Method for Simulation | Matplotlib

### Appy a Simple Haskell Parser — Nov 2022

github.com/JasonWurunfei/SimpleLanguageParserForHaskell

- Appy is a simple parser generator for Haskell programming language. It is a less ambitious version of Happy

Haskell | Functional Programming

### A Simple Stock Exchange Server — Nov 2022

github.com/JasonWurunfei/SimpleStockExchange

- a simple stock exchange application that allows users to open accounts that can hold money and stocks. The fundamental event is that of a trade, which happens when one user has made an offer at the exchange while another user has installed a trader that accepts the given offer.
- The server was implemented fully in Erlang. The testing process was done with the property-based testing approach using QuickCheck

Erlang | Functional programming | Multi-threading | Property-based Testing

## Recognitions & Prize

| | |
|---|---|
| September 2023 | Danish Government Scholarships |
| February 2021 | China National Scholarship |
| December 2019 | Third prize of the National Innovation Design competition of Intelligent Electromechanical Systems for University Students |
| May 2019 | First prize of the Second Chinese University Intelligent Robot Creativity Competition |
| March 2019 | Third prize of the 10th Lanqiao Cup National Software and Information Technology Professionals Competition Jiangsu C/C++ Programming |