Huang Shuangquan

Basic Information

Nationality:	Chinese
Date of Birth:	1989.07
Education:	University of Chinese Academy of Science, Sichuan University
Phone:	+86-17701332388/+65-93882883
Email:	shuangquanhuang@gmail.com
Present Location:	Singapore/China

Work Experience

May. 2021	-	Now	Shopee Singapore	Expert Software Engineer
Dec. 2017	-	May. 2021	Hulu Beijing	Software Engineer
Jun. 2016	-	Nov. 2016	Deepglint Beijing	Senior Software Engineer
Sep. 2015	-	Apr. 2016	Microsoft China	Software Engineer
Jul. 2013	-	Jul. 2015	CA Technologies China	Software Engineer

Education

University Name	Time Period	Major	Degree
University of Chinese Academy of Science	2010.09 - 2013.07	Computer Technique	Master
Sichuan University	2006.09 - 2010.07	Electronic Information	Bachelor

Project Experience

AI Engineering Platform 2022.03-Now

There are 3 annotation platforms we build from zero: audio annotation platform, machine translation platform, and computer assistant translation platform. They are requested from different business teams to tag items and quality checks for better AI performance. At the beginning I owned one of the projects, and then aligned some common architecture designs on different projects especially layered design and BFF like design, to make the project maintainable and easier to develop even if the frontend developer is not familiar with the business logic so well. I also designed and built some common solutions to share with all of the platforms, the basic idea is to dig out the common requirements, split business logic and technical logic, and then design some highly extensible solutions to make them fit different business requests with the same flow and very few code changes, reduced code effort as well as improved the product quality and stability.

As lead of the front end project, except normal development work, I also lead the best practice of the project, including tech stack selection, work split, unified code style tools, component split, CI/CD, git commit and push hooks, interface design and integration with backend side, requirement management, code refact, code review and mentor junior engineers.

Also I extracted some highly reusable components from normal business development to build them as common solution, such as template management, project management, questionnaire component etc. Especially the Questionnaire Component(https://github.com/shuangquanhuang/resume/blob/main/Questionnaire.pdf), which is much more than a component design but a complete solution with high extensibility, and I had cooperated with leaders and PMs to successfully applied into some related projects to proved it's value, and it's has very high potential to extend to company level solutions. Here I'd like to introduce some highlights are:

1. I designed it with these principles: clear boundary (only focus on questionnaire), minimize external dependencies and assumptions about the use environment, extensible schema design, plugin(register your own state machine, components, validators) and interface pattern (use Typescript to ask customized Components/methods implement specific interface) for extension, provide HOOKs and utility functions for user to easily customize.
2. It's designed to have two main Components: Builder and Render, with the Builder user can create questionnaire template schema, and then use Render to render questions.
3. Both Components ares highly extensible, for example: in Builder you can register customized operators, register your own question type, add you own extra settings, use your own validator, register customized input component, register customized validator etc; In Render, question can be render based on DAG and logic expressions, you can register your own single question render component and input component if it follow the interface specification, you can also register you own result validator.
4. It also exports a StateMachine to get the state of current questionnaire, for example, decide whether this question is visible or disabled based on the answer of other questions. You can also register your own state machine to manage the states.
5. Exports lots of utility functions to help user to add customized components, validators, etc.
6. Also wrapped them into grouped questionnaire builder and groupped questionnaire render.

Chatbot data visualization 2021.05-2022.03

Chatbot is an bot to chat with customers for solving some common questions, with the bot we wish the custom service can catch up the increasing of custom numbers, and decrease the number of custom service staff, not only for saving cost but also because we can't hire so many staffs. The visualization platform can provide trend of different metrics, and dive into each metrics to find the reasons, to help operation and teach team to optimize product and strategy to improve customer satisfaction. As the tech lead of frontend project, except normal code work, my work including:

1. Requirement review, tech stack selection, code quality and style management.
2. Build and maintain the development pipeline on CI/CD.
3. Lead best practice for React Component design, including balance between controlled/uncontrolled Component, interaction between Components, function component instead of class component, hooks, error handling, state management, etc.
4. Code style and review, I provided 70% of the code review comments of the project to guide the team members to follow the best practice of React, Typescript etc.
5. Design and implement the version control management tool: Project Loader, with the loader we can

control which version to load in different env, so that we can switch and rollback easily to improve the efficiency.
6. Design and implemented 10+ hooks to reuse the code, especially one hooks to encapsulate network request for unified complex logic such as retry, data formatting and error handling, etc, to reduce bugs.
7. Designed and shared many Components, such as an AdaptiveCardContainer to wrap cards in a variable with container, and contribute it to the company level library.

HULU Web and Chromecast Player 2017.12-2021.05

Hulu is strive to give user the best video experience on different devices, my team owns the video players on web browsers and Chromecast. My work including:

1. Developing new features such as supporting Live/Vod, multiple audio channel, multiple language captions, auto adaptive bitrate, new user interface, supporting different type of Ads, cast, Xlink, 4k and 60 fps, voice command, etc.
2. Project refact, methods including applying better coding style rules for better quality, redesign the architecture by split the project into different layers for better decouple and maintenance, introduce react to replace MVC and jQuery for better development experience and quality.
3. Add a new middle layer between browse site and player to loading player, so that we can release our player independently without cross team work, and we can do phase rollout and dogfooding so that we can figure out quality issues when new version only goes to a small percentage of users to avoid QoS issues.
4. Maintain and update the protocol between Chromecast player and iOS/Android/Web senders, negotiate new requirement and keep back compatible for all of those senders.

Build and optimize QoS metric for player 2017.12-2021.05

QoS is the first priority for HULU since we meet a online accident since 2017, so we did many works to improve our quality including:
1 .Build core QoS metrics, such as video startup time, video startup failure rate, video playback failure rate, rate of exit before video start, connection introduced rebuffering time, etc.

2. Build unified error handling, classify different errors, define unified error code and messaged. And define actions for different errors, for example for some fatal errors we ask player to stop immediately and for some non-fatal errors we ask player to retry several times before stop.
3. Collect QoS data and create dashboard on different tools like Glyph and Coviva to keep track of QoS and start investigate asap when QoS degraded.
4. Utilize big data engines and 3rd party tools like Newrelic, Kibana, Datadog, Presto, etc, to drill down errors to optimize QoS. The general steps including split errors by device model, browser, error type, etc to find out the most suspicious root cause.
5. Integrate those QoS metrics to slack and page duty service to get alert ans respond asap when accident.
6. By defining reasonable QoS metrics and utilizing proper tools, we made our quality better and better continuously, and can make sure our quality will not drop when we release new versions.

Engineering for front end project 2017.12-2021.05

We maintained many different projects and all of them need CI/CD to build and release easily, and some of our projects need functions like encrypt, Ads, release history, etc, those requirements need us to maintain and upgrade development and test tools to let us develop easily and release high quality projects. The works including:

1. Upgrade Gulp to Webpack, with new tool chain we can support our development better and keep our daily life easier.
2. Maintain virtual machine cluster, build Docker image, maintain Jenkins, etc, to build , test and release automatically.
3. Develop release history management tools so that we and other relative teams can easily track our release status, which is very useful for other teams to do a quick check when they see some functions broken.
4. Maintain tools for Ads debugging for example encrypt and decrypt Ads request and response, get specific Ads for debugging, etc.
5. Better tools and practice can not only enable us to release high quality code and product, but also can make our daily life happier to become more productive and creative.

Pipeline and Baseline for content based video encoding 2017.12-2018-05

The encoding parameters for video encoding are basically based on experienced engineer, general strategy is using different group of parameters on different type of video for example cartoon, action, comedy, etc. And machine leaning has been proved can do better than human on many area after trained on large scale of data, so we want to build a model to guess the best encoding parameter for each of different single video to improve our video quality within the same bandwidth. And I built a pipeline to extract huge video feature on machine cluster, and store those features and using xgboost to get a basic model as baseline to help researcher to training a model, the result proved it can gain about 5% data compression on some videos.

Big data platform for video 2016.07-2016.11

It’s a core and competitive technology of Deepglint to search target in huge amount of videos, the platform extract features from videos to generate structural data and provide interface for users to do their business for example search routine of a car by it’s plate number of even a small part photo of it. I worked as a full stack engineer to write REST API with Go language and also develop front end project with VUE.

Data analysis on Office 365 2015.09-2016.05

Office 365 is a SaaS version of Office runs on a huge cloud, there are many services and our team owns network availability for it for example availability of DNS, data center, top of rack, and even single machine. My daily work including analyzing log data on Cosmos to provide dashboard, alert on different level, track errors and QoS and cooperate with relative teams to fix issues.

Instrument tools for Java application 2013.07-2015.07

Java agent is a technology that you can insert prob into Java application to get the runtime code stack frame to analyze application performance and errors. We built a Java agent to collect, filter, classify and integrate information from application to show system architecture, performance bottleneck and key nodes to users, so that user can understand low level details and then locate root cause and evaluate test plan. And we can also automatically provide automation test cases and virtual services to users. My works including analyzing different protocols like JMS, Servlet, Rest and WebService etc, and provide relationship between them and provide REST API for them; provide API to create automation test cases and virtual services for specific node; provide API to search runtime status; generate complicate report; analyze memory and cpu bottleneck; utilizing search engine like ELK to improve search performance.

Related Links

Github https://github.com/shuangquanhuang

Personal Website https://shuangquanhuang.github.io/

Algorithms https://github.com/shuangquanhuang/algorithms

LeetCode https://leetcode-cn.com/u/firepaw/

A simple project to show react https://github.com/shuangquanhuang/agorademo

Self Description

I have very good basic computer science knowledge including very strong algorithm and design skills, very strong learning quality.

I have both backend frontend software development experience, have good understanding about how to complete project well.

I always ask for very high code quality, my code are always very clear and easy to maintain, have strong quality assurance for the products I delivered.

I have deep understanding of the the tech I used, be able to keep learning, keep track for the new techniques.

I like simple and reliable environment, where we always make sure the product we delivered are fully tested to keep our reputation as well as to reduce ineffective communication, to improve the overall efficiency of the team.

Other Information

Familiar with many coding language like Java, Python, Javascript, Typescript, Go, etc.

Familiar with agile development.

Familiar with architecture design, object oriented design and design patterns, MVVM.

Familiar with database like MySql and SQL language.

Familiar with lots of algorithm and data structure, rank 100+ on Leetcode.