

# ANTHONY MAI

0434 892 096 | [Anthony.Mai@hotmail.com](mailto:Anthony.Mai@hotmail.com) | [Anto.au](http://Anto.au) | Melbourne, Australia

## INDUSTRY EXPERIENCE

---

### Software Engineer

2021 - 2023

Slalom Build

Part of a team responsible for uplifting the Small Business Banking system and streamlining the process of creating loans. Most recently, worked on and launched ANZ's new lending system (ANZ GoBiz) and ensured its quality through test automation.

#### *Outcomes and Responsibilities*

- Migrated over 400 test cases from LISA (CA DevTest) to a separate testing framework using JavaScript, Chai, Mocha and SuperTest - in addition to migrating several virtualised services to Mountebank
- Managed several platform releases as the QA lead, leading a group of QA engineers
- Uplifted QA legacy documentation to living Confluence pages - making it more readily accessible and manageable by the team
- Developed a UI automation suite using Puppeteer and Cucumber.js for ANZ GoBiz, guaranteeing software quality and allowing risk-free feature development

### QA/Automation Engineer

2019 - 2021

ANZ Banking Group Ltd (Cognizant Technology Solutions)

Part of a team responsible for uplifting the Small Business Banking system and streamlining the process of creating loans. Most recently, worked on and launched ANZ's new lending system (ANZ GoBiz) and ensured its quality through test automation.

#### *Outcomes and Responsibilities*

- Migrated over 400 test cases from LISA (CA DevTest) to a separate testing framework using JavaScript, Chai, Mocha and SuperTest - in addition to migrating several virtualised services to Mountebank
- Managed several platform releases as the QA lead, leading a group of QA engineers
- Uplifted QA legacy documentation to living Confluence pages - making it more readily accessible and manageable by the team
- Developed a UI automation suite using Puppeteer and Cucumber.js for ANZ GoBiz, guaranteeing software quality and allowing risk-free feature development

### Associate Software Engineer

2018 - 2021

Cognizant Technology Solutions

- Participated in the graduate programme in which I was part of a team that created an online store using React and Redux for the front-end that interacted with a microservice built using Java and Spring boot.

### Analyst Programmer

2015 - 2017

Synchronised Software Pty Ltd

- Analysed requirements, working closely with business analysts to identify gaps and ambiguity in the functional specifications to reduce the number of potential defects
- Became proficient in SyncSoft's proprietary language - Capital Translator Language (CTL), going on to mentor new analyst programmers
- Followed development procedures involving source control, implementing technical specifications and participating in code reviews
- Implemented new modules and updates for SyncSoft's Capital registry administration system consisting of forms and interfaces that interact with various databases
- Responsible for developing complex CTL functions and SQL procedures and transactions

## CERTIFICATIONS

**SAFe 4 Practitioner**

Nov 2018

Scaled Agile, Inc.

## EDUCATION

**Bachelor of Computer Science**

Feb 2014 - Nov 2017

Swinburne University of Technology

## TECHNICAL SKILLS

- **Languages:** JavaScript, HTML, CSS, C#, SQL, Java
- **Frameworks and Libraries:** Mocha, Chai, SuperTest, Cucumber.js, Puppeteer, ReactJS, Jest, Spring Boot
- **Others:** NodeJS, Git, Confluence, Jira, SQL Server, Mountebank, Splunk

Things to add:

- Equifax project
  - Building Java Springboot microservices
  - Working with GCP Firestore (NoSQL DB), Pub/Sub, Bigtable
  - OpenAPI for API specs
- BP SuperFleet
  - Micro frontend architecture
    - Each MFE has an entry point, exposes what it likes, has remotes where it downloads types and uses other exposed elements of other MFEs
  - Webpack incl. Module Federation
  - CRACO (overriding config with our own plugins (TypeScript federation and Module Federation))
  - React, TypeScript, JavaScript
  - Cypress
  - AWS (CDK stacks, S3 buckets, Route53, CloudFront)
  - Performance improvements via cache control, HTTP3
  - Challenges with MFE architecture
    - Shared dependencies (impacts to performance)
    - Separate AWS accounts, deployments, pipelines
    - One MFE affecting another due to implicit coupling or it being down
    - Error boundaries
    - Fallbacks when Webpack isn't able to mount a container containing an MFE
    - Fallbacks when there are chunk load errors (remove shared dependencies)

- CSS leaks
- State management