

Statement of Purpose

Zhenghang Hu

After three years of searching during college, I decided on databases as my future research direction in the Ph.D. program. The decision was driven by taking CMSC624, *Database System Architecture and Implementation*, and my current teaching assistantship in CMSC424, *Database Design*, at the University of Maryland. Out of class, two other experiences further prepared me for research. My ACM ICPC(International Collegiate Programming Contest) experiences established my solid algorithms and data structures foundations. My start-up experience allowed me to apply database design theories to a real-world application.

Since high school, I have been doing competitive programming, and I believe my algorithm skills gained from contest training have provided me with a solid base of critical thinking and research abilities. In 2018, I joined the third ACM team at UMD to maintain the intensive training. In 2019, I was selected into the first team and competed in the ACM ICPC Mid-Atlantic USA Regional Contest. We were ranked **3rd** among the 154 teams sent by 53 universities in the Mid-Atlantic region and made it to the 2020 ICPC North America Championship. Most importantly, training for ICPC strengthened my perseverance for solving problems and determination to comprehensively study an algorithm because hard problems often require a whole night of sitting, and a shallow understanding of an algorithm is inadequate for its implementation and varied applications.

In the summer of 2021, a classmate and I founded a start-up called ShellBay. Its primary goal is to develop the namesake app that provides an online marketplace for college students. We built up the app from scratch within the summer, and now we are in the process of redesigning the app with our UI/UX team. My responsibilities span the entire backend architecture, the optimal database schema design, interactions with the database, and essential frontend user interfaces. During the process, I drew the ER diagram, converted it into relational schemas, and applied appropriate decompositions to ensure the schemas are in BCNF to reduce the redundancy to the slightest degree. To study required technologies such as NodeJS, React Native, Firebase, and Firestore (a NoSQL cloud database), every day, I uninterruptedly researched their documentation, source codes, community support, and APIs. Through applying the common design patterns within the industry, I gained a bigger picture of how popular frameworks are engineered, how an application is architected, and how a good schema can boost performance in a real case.

This fall semester, I am taking CMSC624, a graduate-level course focused on database architectures. Papers and projects introduced in this course further raised my interest in databases and allowed me to explore database design deeper. I implemented five classic concurrency control schemes for a database stored in the main memory in this class. Through testing those

schemes under simulations of varied use cases, I got a clear view of the trade-off in each scheme and under what scenario each scheme can achieve the highest throughput. Apart from the projects, I learned seminal research dedicated to different database components that enhanced my understanding of the overall architecture of DBMS. For example, OCC offered me a perspective of an optimistic approach to execute transactions and decide on commit or abort through validation, avoiding the locking overhead; through DBMIN, I learned that database operations, based on their reference patterns, can be associated with the most appropriate page replacement strategies; With the learning of ARIES, I understood how to preserve atomicity and durability under system crash via physiological logging and Write-Ahead Logging protocol. As I went further to study the state-of-art distributed databases, I was more aware of the complexities and challenges brought by modern DBMS, such as the need to upgrade serializability to strict serializability to avoid anomalies and the need for 2PC protocol to ensure the atomicity of global transactions.

The current explosive data growth requires more fault-tolerant and highly scalable database systems. It's an intriguing and not-fully-answered question about how we should architect databases to trade-off between correctness, performance, availability, etc., in the most reasonable way. With my passion for computer science, keen interest in databases, and my strong perseverance in solving problems, I long to be a qualified researcher at the University of Wisconsin-Madison. And I sincerely hope the acceptance of UW-Madison's Ph.D. program could be a milestone on my journey to become a database scientist.