# Daniel Pavenko

linkedin.com/in/dan-pavenko
danpavenko.com

Email: pavenkodanielofficial@hotmail.com
GitHub: github.com/Danpav1
Mobile: (704)-777-4104

## Technical Skills

**Languages**: Java, JavaScript, Python, C, SQL, Bash, HTML, CSS, LaTeX, Russian, English

**Frameworks & Libraries**: React, Node.js, Express, Sequelize ORM, React Router DOM, Axios, React Context API, Tailwind CSS, Bootstrap, Swing

**Tools & Technologies**: Git (GitHub, GitLab), Vite, VS Code, Eclipse, IntelliJ, CLion, RegEx, Valgrind, Linux, Unix

## Education

**Shippensburg University of Pennsylvania** — Shippensburg, PA
*Bachelor of Science in Computer Science* — *Jan. 2021 – Dec. 2025 (Expected)*

- GPA: 3.684; Honors & Awards: Dean's List for all semesters

## Professional Experience

**Schreiber Foods** — Shippensburg, PA
*Technology Intern* — *Aug 2023 - Present*

- Implemented Agile methodologies, including Scrum and Kanban, to streamline project workflows and enhance team collaboration.
- Managed and maintained server infrastructure and Oracle databases supporting operations for 600+ employees.
- Deployed and configured enterprise hardware, ensuring seamless IT operations across the plant.

## Projects

**User Authentication System**
https://github.com/Danpav1/login_website
*React, React Router DOM, React Context API, Vite, Tailwind CSS, Axios, Node.js, Express, Sequelize ORM, SQLite, JWT, bcryptjs, dotenv*

- Developed a secure user login and registration platform using JWT and bcryptjs for authentication.
- Built a responsive frontend with React and Tailwind CSS, ensuring seamless user experience across devices.
- Implemented backend services with Node.js, Express, and Sequelize ORM to manage user data in a SQLite database.
- Delivered a sleek full-stack web application with seamless integration of login and registration features for secure user management.

**Process Shell**
https://github.com/Danpav1/shell/tree/main
*C, Bash, Valgrind, Git*

- Engineered a custom Unix shell in C with features like scripting, I/O redirection, and parallel command support.
- Leveraged multi-processing to enable concurrent command execution, enhancing performance.
- Developed a testing harness using TDD methodologies to ensure software reliability and robustness.
- Used Valgrind to identify and resolve memory leaks, ensuring efficient memory usage.