**Bogdan Shmat**

**Phone:** (425) 319-5562 **Email:** [bogdan.shmat@wsu.edu](mailto:bogdan.shmat@wsu.edu) **Address:** 6228 146th PL SE Everett WA, 98208

**Linkedin:** <https://www.linkedin.com/in/bogdan-shmat-b42552222/> **Website:** <https://bogdansportfolio.azurewebsites.net/>

**OBJECTIVE**

**Seeking for Software Development Engineer Position**

**SKILLS**

**Computer Skills:** TypeScript/JavaScript, React, Python, C#, Java, Node.js, SQL, Django, Next.js

**Tools:** Git, Azure Dev Ops, Visual Studio Code, Gitlab, CICD

**Soft Skills:** Teamwork, communication skills, patience, enthusiasm

**EDUCATION**

**Washington State University** Aug 2020-May-2024

BS in Software Engineering

**EXPERIENCE**

**Frontend Developer Intern| Invent | Remote**

* Assigned tickets through Jira and clarified ambiguous requirements with the corresponding project manager to ensure ticket success.
* Completed high impact tasks such as a high complexity addition into the client facing UI library using Styled-Components, TypeScript, React, React Testing library and Jest.

**Projects**

Log meals and workouts | [**https://fittrack.azurewebsites.net/**](https://fittrack.azurewebsites.net/) **| TypeScript, React, Next.js, MySQL**

* Used Tailwind.css to create forms for quick add, add from user’s collection, and add via Nutritionix endpoint with similar CRUD operations for exercises.
* Authentication and secure routes are handled using Clerk.
* Hosted on Azure App Services with database configured through PlanetScale.

**Kotlin version using Room and BCrypt |** [**https://github.com/boshma/FitnessAppKotlin**](https://github.com/boshma/FitnessAppKotlin)

**Checkers |** [**https://github.com/boshma/Checkers-Kotlin-**](https://github.com/boshma/Checkers-Kotlin-) **| Kotlin**

* Follows complete rule set of checkers, users play on one device, can see the score, and reset game.

**Discord Clone |** [**https://github.com/boshma/FitnessCommunity**](https://github.com/boshma/FitnessCommunity) **| Python, Django, CSS, JavaScript, SQLite**

* CRUD operations for rooms, topics, and chat room messages.
* User Creation and Authentication implemented using Django’s native functionality.