**# Ghost Net Fishing**

„Ghost Net Fishing“ ist eine Webanwendung, die im Rahmen einer Fallstudie im Studiengang Informatik an der IU Internationale Hochschule entwickelt wurde. Ziel des Projekts ist es, die digitale Erfassung und Bergung sogenannter Geisternetze zu ermöglichen. Diese herrenlosen Fischernetze treiben in den Weltmeeren und stellen eine erhebliche Gefahr für Meereslebewesen, Umwelt und Schifffahrt dar.

Die Anwendung richtet sich an meldende und bergende Personen, die Geisternetze mit Standortdaten und geschätzter Größe melden, deren Status aktualisieren und sich für Bergungen registrieren können. Meldungen können optional anonym erfolgen. Der Status eines Netzes lässt sich in mehreren Schritten anpassen – von „GEMELDET“ über „IN BERGUNG“ bis „GEBORGEN“ oder „VERSCHOLLEN“. Ziel ist es, eine benutzerfreundliche, funktional fokussierte Oberfläche ohne Loginpflicht bereitzustellen, die gleichzeitig zuverlässige Datenhaltung über eine relationale Datenbank ermöglicht.

Die Softwarearchitektur basiert auf dem MVC-Muster. Die Geschäftslogik wird im Backend durch Spring Boot, CDI und JPA (Hibernate) realisiert. Die Daten werden dauerhaft in einer MySQL-Datenbank gespeichert. Die Anwendung unterstützt eine einfache Interaktion über JSF-gestützte Benutzeroberflächen.

**Technologien**

- \*\*Frontend:\*\* HTML, JSF (JavaServer Faces), ggf. PrimeFaces

- \*\*Backend:\*\* Spring Boot, Java, CDI, JPA (Hibernate)

- \*\*Datenbank:\*\* MySQL

- \*\*Architektur:\*\* MVC (Model-View-Controller)

**Datenmodell (Kurzbeschreibung)**

- \*\*Geisternetz:\*\* ID, Standort (Koordinaten), Größe, Status, meldende/bergende Person

- \*\*Person:\*\* ID, Name, Telefonnummer

- \*\*Bergung:\*\* Beziehung zwischen Geisternetz und bergender Person

**Voraussetzungen**

- Java 17 oder neuer

- Maven

- Laufender MySQL-Server

- Webbrowser (z. B. Chrome oder Firefox)

**Setup und Ausführung**

Um die Anwendung lokal auszuführen, benötigt man eine Java-Entwicklungsumgebung (mindestens Version 17), das Build-Tool Maven sowie einen laufenden MySQL-Server. Nachdem das GitHub-Repository geklont wurde (`git clone`), wechselt man in das Projektverzeichnis und führt den Befehl `mvn clean install` aus, um die Anwendung zu kompilieren und alle benötigten Abhängigkeiten herunterzuladen.

Anschließend sollte die MySQL-Datenbank gestartet und – falls nötig – ein bereitgestelltes SQL-Skript aus dem Verzeichnis `/db` ausgeführt werden, um die nötigen Tabellen und Startdaten anzulegen. Sobald die Datenbank bereitsteht, kann die Anwendung mit `mvn spring-boot:run` gestartet werden. Nach erfolgreichem Start ist die Weboberfläche über [http://localhost:8080] (http://localhost:8080) im Browser erreichbar.

**GitHub-Repository**

Das vollständige Projekt ist auf GitHub verfügbar:

[https://github.com/GvendaHemel/GhostNetFishing] (https://github.com/GvendaHemel/GhostNetFishing)

**Autorin**

Dieses Projekt wurde im Rahmen einer Fallstudie an der IU Internationale Hochschule erstellt von:

\*\*Gvantsa Gogrichiani\*\*

GitHub: [@GvendaHemel](https://github.com/GvendaHemel)