DailyTracker – Technischer Start & Erste Umsetzungsschritte

# 1. Ziel dieses Dokuments

Dieses Dokument beschreibt die ersten technischen Umsetzungsschritte für das Projekt 'DailyTracker'. Es enthält eine Anleitung zum Erstellen des Projekts in IntelliJ IDEA mit Spring Boot, dem Setup der ersten Klassen, Packages und grundlegender Projektstruktur.

# 2. Projektstruktur (Backend – Java Spring Boot)

src/  
├── main/  
│ ├── java/com/dailytracker/  
│ │ ├── controller/  
│ │ ├── model/  
│ │ ├── repository/  
│ │ └── service/  
│ └── resources/  
│ └── application.properties  
└── test/  
 └── java/com/dailytracker/

# 3. Projekt in IntelliJ erstellen

- IntelliJ IDEA starten  
- Neues Projekt > Spring Initializr wählen  
- Group: com.dailytracker  
- Artifact: dailytracker  
- Dependencies: Spring Web, Spring Data JPA, PostgreSQL Driver, Spring Boot DevTools, Lombok  
- Projekt erstellen und öffnen

# 4. Erste Java-Klassen

Package: com.dailytracker.model

public class DailyEntry {  
 private Long id;  
 private LocalDate datum;  
 private int schritte;  
 private int schlafMinuten;  
 private double wetterTempNacht;  
 private String mondphase;  
}

Package: com.dailytracker.model

public class SupplementEntry {  
 private Long id;  
 private String name;  
 private double mengeMg;  
 private boolean genommen;  
}

# 5. Repository Interfaces

public interface DailyEntryRepository extends JpaRepository<DailyEntry, Long> {}  
public interface SupplementEntryRepository extends JpaRepository<SupplementEntry, Long> {}

# 6. Erster Controller (Beispiel)

@RestController  
@RequestMapping("/api/daily")  
public class DailyEntryController {  
 @Autowired  
 private DailyEntryRepository dailyRepo;  
  
 @PostMapping  
 public DailyEntry createEntry(@RequestBody DailyEntry entry) {  
 return dailyRepo.save(entry);  
 }  
}

# 7. Anwendung starten

- PostgreSQL lokal installieren oder Docker nutzen  
- application.properties anpassen mit DB-Zugangsdaten  
- Projekt mit Spring Boot starten  
- POST-Request via Postman an http://localhost:8080/api/daily senden