### **Kravspecifikation for "Hjælp! Jeg er en fisk?" App**

### **Introduktion**

"Hjælp! Jeg er en fisk?" er en innovativ fitnessapp målrettet mod folk, der ikke normalt går i fitnesscenter. Målet er at gøre træning sjov, tilgængelig og motiverende for alle, uanset deres udgangspunkt. Dette dokument beskriver de detaljerede krav til udviklingen af appen ved hjælp af Node.js som backend, Flutter til frontend og MongoDB som database.

### **Teknologier**

* **Backend:** Node.js
  1. Brug af Express.js til opbygning af RESTful API'er.
  2. Middleware til autentificering og autorisation.
  3. Integration med MongoDB for datahåndtering.
* **Frontend:** Flutter
  1. Udvikling af en responsiv brugergrænseflade.
  2. Implementering af navigationsstrukturer
  3. Brug af Flutter packages for kort, grafik og real-time opdateringer.
* **Database:** MongoDB
  1. Design af skemaer for brugere, træningsdata og konkurrencer.
  2. Indeksering for hurtig forespørgselshåndtering.
  3. Brug af Mongoose til data validering og modellering.

### **Funktionaliteter**

### **Geo-lokation**

1. **Sporing af ruter**
   * Brug GPS til at spore ruter, hastighed og tid.
     1. Implementering af real-time GPS-sporing.
     2. Gemme rutedata i databasen.
     3. Vise ruten på et kort i appen.
   * Rutedata skal inkludere distance, tid, gennemsnitlig hastighed
     1. Beregning af distance baseret på GPS-koordinater.
     2. Tidsregistrering for hver rute.
   * Brugerne skal kunne gemme og gennemgå deres ruter i en historik.
     1. Oprettelse af en historikside i appen.
     2. Mulighed for at filtrere og søge i historik.
     3. Vise detaljerede oplysninger om hver rute.
2. **Indtjekning**
   * Automatisk indtjekning ved ankomst til fitnesscenteret ved hjælp af GPS.
     1. Geofence opsætning omkring fitnesscentre.
     2. Push-beskeder ved indtjekning.
     3. Logføring af indtjekningstidspunkter.
   * Appen skal kunne identificere hvilket center brugeren befinder sig i og vise relevante oplysninger.
     1. Liste over fitnesscentre med deres GPS-koordinater.
     2. Visning af centeroplysninger som åbningstider og faciliteter.
     3. Vise aktuelle events eller træningssessioner i centeret.

### **Planlægning**

1. **Træningsplanlægning**
   * Brugere skal kunne oprette og gemme træningsplaner for fremtidige træningspas.
     1. Formular til oprettelse af træningsplaner.
     2. Mulighed for at gemme og redigere planer.
     3. Kalenderintegration til planlægning.
   * Planer kan inkludere specifikke øvelser, gentagelser, sæt, og mål (f.eks. tid, vægt).
     1. Vælg øvelser fra en database.
     2. Angivelse af gentagelser og sæt for hver øvelse.
     3. Mulighed for at sætte mål for hver session.
   * Inviter andre brugere til at deltage i planlagte træningspas via notifikationer eller delingsfunktioner.
     1. Notifikationssystem til inviteringer.
     2. Deling af planer via sociale medier.
     3. Push-beskeder til inviterede brugere.
2. **Personlige træningsplaner**
   * Brugere skal kunne oprette og tilpasse personlige træningsplaner baseret på deres mål og præferencer.
     1. Initial spørgeskema for at identificere brugerens mål.
     2. Automatisk generering af forslag til træningsplaner.
     3. Mulighed for manuel tilpasning af planer.
   * Appen skal kunne give anbefalinger til træningsøvelser baseret på brugerens historik og mål.
     1. Analyse af brugernes træningshistorik.
     2. Anbefalinger baseret på brugerens præstationer og fremskridt.
     3. Mulighed for at gemme favoritøvelser.

### **Analyse**

1. **Træningsstats**
   * Brugere skal kunne se detaljerede statistikker om deres træning, herunder tid brugt, kg løftet, og kalorier forbrændt.
     1. Opsummering af hver træningssession.
     2. Visualisering af statistikker i grafer.
     3. Sammenligning af præstationer over tid.
   * Statistikker skal kunne vises som grafer og tabeller for bedre visualisering.
     1. Linje- og søjlediagrammer for at vise fremskridt.
     2. Data-tabeller for detaljeret visning.
     3. Mulighed for at eksportere data.
   * Sammenligning af stats fra forskellige træningspas og over tid.
     1. Filtrering af data baseret på datoer.
     2. Sammenligningsfunktion for forskellige tidsperioder.
     3. Historisk analyse for at se langtidstrends.
2. **Historik**
   * En detaljeret historik af alle træningspas, inklusive dato, tid, deltagere og præstationer.
     1. Logføring af hver træningssession.
     2. Visning af deltagerliste for gruppeaktiviteter.
     3. Noter og kommentarer for hver session.
   * Brugere skal kunne filtrere og søge i deres historik for at finde specifikke træningspas eller resultater.
     1. Søgning baseret på dato og type af træning.
     2. Filtrering efter deltagere og mål.
     3. Sorteringsmuligheder for nem visning.

### **Ruteplanlægning**

1. **Planlægning af ruter**
   * Brugerne skal kunne planlægge løbe- og cykelruter på forhånd.
     1. Kortintegration til ruteplanlægning.
     2. Mulighed for at gemme og redigere ruter.
     3. Deling af ruter med andre brugere.
   * Integration med korttjenester (f.eks. Google Maps) for ruteplanlægning og visning.
     1. Brug af Google Maps API for nøjagtig kortvisning.
     2. Ruteoptimering baseret på brugerens præferencer.
     3. Real-time opdatering af ruter.
   * Inviter andre brugere til at deltage i planlagte ruter via notifikationer eller delingsfunktioner.
     1. Notifikationssystem til inviteringer.
     2. Deling af ruter via sociale medier.
     3. Push-beskeder til inviterede brugere.

### **Deling**

1. **Træningscyklusser**
   * Mulighed for at dele træningscyklusser og resultater med andre brugere.
     1. Deling via sociale medier og direkte i appen.
     2. Mulighed for at sende invitationer til venner.
     3. Integration med e-mail og beskedapps.
   * Brugere kan dele deres planer, resultater og statistik på sociale medier eller direkte via appen.
     1. Indbygget delingsfunktion.
     2. Forudindstillede delingsskabeloner.
     3. Automatiske opdateringer og delinger.
2. **QR-kode scanning**
   * Brug kameraet til at scanne QR-koder for hurtigt at dele aktivitet eller deltage i konkurrencer.
     1. QR-kode generator for aktiviteter.
     2. QR-kode scanner i appen.
     3. Logføring af delte aktiviteter.
   * QR-koder kan genereres for specifikke aktiviteter eller konkurrencer og deles med andre.
     1. Oprettelse af unikke QR-koder for hver aktivitet.
     2. Deling af QR-koder via e-mail og beskedapps.
     3. Visning af delte QR-koder på brugerens profil.

### **Konkurrence**

1. **Oprettelse af konkurrencer**
   * Brugerne kan oprette konkurrencer med specifikke mål (f.eks. hvem går mest på en uge, hvem løfter flest kg).
     1. Konkurrenceformular for at oprette nye konkurrencer.
     2. Definere mål og regler for konkurrencen.
     3. Mulighed for at invitere deltagere via appen.
   * Konkurrencer kan være åbne for alle eller begrænset til specifikke brugere eller grupper.
     1. Indstilling af privat eller offentlig konkurrence.
     2. Invitation og godkendelse af deltagere.
     3. Administration af konkurrencer og deltagere.
2. **Gruppekonkurrencer**
   * Mulighed for at oprette grupper og konkurrere mod hinanden inden for gruppen.
     1. Oprettelse af brugergrupper.
     2. Invitering af brugere til grupper.
     3. Administration af gruppemedlemmer og konkurrencer.
   * Brugere kan invitere venner eller kolleger til at deltage i gruppens konkurrencer.
     1. Deling af gruppeinvitationer via sociale medier.
     2. Push-beskeder til inviterede brugere.
     3. Notifikationssystem for gruppens aktiviteter.
3. **Globale konkurrencer**
   * Ugentlige og månedlige globale konkurrencer, f.eks. hvem har løbet flest km.
     1. Automatisk oprettelse af globale konkurrencer.
     2. Visning af globale konkurrencer i en dedikeret sektion.
     3. Belønningssystem for vindere.
   * Ranglister, der viser brugernes position i globale konkurrencer.
     1. Real-time opdatering af ranglister.
     2. Visning af top-deltagere og deres præstationer.
     3. Notifikationer om brugerens placering og fremskridt.

### **Brugerprofil**

1. **Profilbillede**
   * Brugerne skal kunne uploade et profilbillede, som vises på deres profil og i konkurrencer.
     1. Mulighed for at tage billede med kamera eller vælge fra galleri.
     2. Redigeringsværktøjer til beskæring og justering af billeder.
     3. Sikring af billedkvalitet og størrelse.
2. **Kontaktinformation**
   * Angivelse af fuldt navn, telefonnummer og e-mailadresse.
     1. Validering af e-mail og telefonnummer.
     2. Sikring af privatliv ved at skjule kontaktoplysninger.
     3. Mulighed for at ændre kontaktoplysninger.
3. **Brugernavn**
   * Brugernavn til tilføjelse af venner og deltagelse i konkurrencer.
     1. Validering af unikke brugernavne.
     2. Visning af brugernavn på profil og i konkurrencer.
     3. Mulighed for at ændre brugernavn.

### **Ikke-funktionelle krav**

### **Skalerbarhed**

* Systemet skal kunne håndtere en stor mængde brugere og data uden performance degradation.
  1. Brug af cloud-tjenester for skalerbar infrastruktur.
  2. Lastbalancering for at sikre ydeevne under høje belastninger.
  3. Horisontal og vertikal skalerbarhed af servere.

### **Sikkerhed**

* Sikring af brugerdata gennem kryptering af data i hvile og under transport.
  1. Brug af HTTPS til datatransmission.
  2. AES-kryptering for data i hvile.
  3. Regelmæssig sikkerhedsrevision.
* Implementering af sikre autentificeringsmetoder, såsom to-faktor autentificering.
  1. Integration af to-faktor autentificering.
  2. Stærk password policy.
  3. Brug af OAuth 2.0 for tredjeparts integrationer.

### **Tilgængelighed**

* Appen skal være tilgængelig på alle platforme (iOS og Android) med ensartet brugeroplevelse.
  1. Ensartet design på tværs af platforme.
  2. Support for forskellige skærmstørrelser og opløsninger.
  3. Regelmæssige opdateringer for at sikre kompatibilitet.

### **Performance**

* Hurtig og responsiv brugergrænseflade med minimal indlæsnings- og ventetid.
  1. Optimering af frontend-kode.
  2. Brug af caching for at reducere indlæsningstid.
  3. Optimering af databaseforespørgsler.

### **Vedligeholdelse**

* Koden skal være modulær og let at vedligeholde og udvide med nye funktioner.
  1. Brug af modulær arkitektur.
  2. Dokumentation af kode og systemarkitektur.
  3. Regelmæssig refaktorisering af kodebase.

### **API'er og Integrationer**

1. **Geo-lokation API**
   * Brug af GPS til sporing og indtjekning.
     1. Implementering af real-time GPS-sporing.
     2. Integration med tredjeparts korttjenester.
     3. Optimering af GPS-brug for at spare batteri.
2. **Push-beskeder**
   * Integration af push-notifikationer for træningspåmindelser, konkurrenceopdateringer osv.
     1. Brug af Firebase Cloud Messaging.
     2. Konfiguration af push-notifikationer for forskellige begivenheder.
     3. Mulighed for at tilpasse notifikationsindstillinger.
3. **Sociale medier**
   * Integration med sociale medier som Facebook, Instagram og Twitter til deling af træningsresultater og konkurrencer.
     1. Brug af OAuth til sikker autentificering.
     2. Mulighed for at poste resultater direkte fra appen.
     3. Integration af sociale medie widgets i brugerprofiler.

### **Design og Brugeroplevelse**

### **Brugergrænseflade**

* Intuitiv og brugervenlig UI/UX design, der gør det nemt for brugerne at navigere og bruge appens funktioner.
  1. Brugercentreret designproces.
  2. Feedback loops for kontinuerlig forbedring.
  3. Brug af velkendte designmønstre.

### **Onboarding**

* En simpel og effektiv onboarding-proces for nye brugere.
  1. Interaktive tutorials.
  2. Trin-for-trin guide til opsætning af profil.
  3. Mulighed for at springe onboarding over og vende tilbage senere.

### **Implementeringsplan**

### **Fase 1: Design og Prototyping**

* Udarbejdelse af design mockups og prototyper.
  1. Skabelse af wireframes.
  2. Brugerfeedback på prototyper.
  3. Iterativ forbedring baseret på feedback.
* Brugerundersøgelser og feedback for at sikre, at designet opfylder brugernes behov.
  1. Fokusgrupper.
  2. Online spørgeskemaer.
  3. A/B-testning af designløsninger.

### **Fase 2: Backend-udvikling**

* Opsætning af Node.js backend og MongoDB database.
  1. Konfiguration af servermiljø.
  2. Design og implementering af API-endepunkter.
  3. Opsætning af databaseforbindelser og skemaer.
* Implementering af API'er for geo-lokation, træningsplanlægning, konkurrencer osv.
  1. Udvikling af geo-lokationsservice.
  2. Oprettelse af træningsplan API.
  3. Konkurrenceadministrationssystem.

### **Fase 3: Frontend-udvikling**

* Udvikling af Flutter-app med alle planlagte funktioner.
  1. Udvikling af hovedgrænsefladen.
  2. Integration af backend-API'er.
  3. UI/UX testning og optimering.
* Integration med backend-API'er.
  1. Opsætning af API-klient i Flutter.
  2. Håndtering af netværksanmodninger.
  3. Fejlhåndtering og brugerfeedback.

### **Fase 4: Integration og Testning**

* Integration af frontend og backend.
  1. End-to-end testning.
  2. Integrationstest for at sikre sammenhæng mellem frontend og backend.
  3. Belastningstest for at sikre ydeevne under høj belastning.
* Grundig testning af appen for at sikre kvalitet og stabilitet.
  1. Enhedstestning.
  2. Systemtestning.
  3. Brugeraccepttestning (UAT).

### **Fase 5: Lancering og Vedligeholdelse**

* Officiel lancering af appen på iOS og Android platforme.
  1. Markedsføringsstrategi for lancering.
  2. Opsætning af supportkanaler.
  3. Indsamling af brugerfeedback efter lancering.
* Markedsføring og brugerengagement for at tiltrække nye brugere.
  1. Kampagner på sociale medier.
  2. Influencer samarbejder.
  3. Events og konkurrencer for brugerengagement.
* Løbende vedligeholdelse og opdatering baseret på brugerfeedback og teknologiske fremskridt.
  1. Regelmæssige opdateringer og bug fixes.
  2. Introduktion af nye funktioner baseret på feedback.
  3. Kontinuerlig overvågning af systemets ydeevne og sikkerhed.

### **Konklusion**

Denne detaljerede kravspecifikation danner grundlaget for udviklingen af "Hjælp! Jeg er en fisk?" appen. Ved at følge denne specifikation sikrer vi, at appen lever op til både funktionelle og ikke-funktionelle krav, og skaber en positiv brugeroplevelse, der motiverer og engagerer brugerne i deres fitnessrejse.