NODE JS, TELEGRAF. JS VE MONGO DB KULLANARAK TELEGRAM CHATBOT YAPIMI

Hakan AKDOĞAN 19360859016

SUNUM PLANI

Birinci Kısım

ChatBot nedir? Hangi Alanlarda Kullanılır

İkinci Kısım

Yabancı Dil Eğitimi Alanında ChatBot Kullanımı

Üçüncü Kısım

Yapılan Projenin Tanıtımı

Dördüncü Kısım

Projede Kullanılan Teknolojilerin Tanıtımı

Beşinci Kısım

Projeden Kod Örnekleri

Altıncı Kısım

Uygulama İçeriği

CHATBOT NEDIR?

- ChatBotlar birer bilgisayar programıdır
- Kullanıcılara hizmet için vardır
 - İnsan Yükünü Azaltır



CHATBOTLARIN KULLANIM ALANLARI





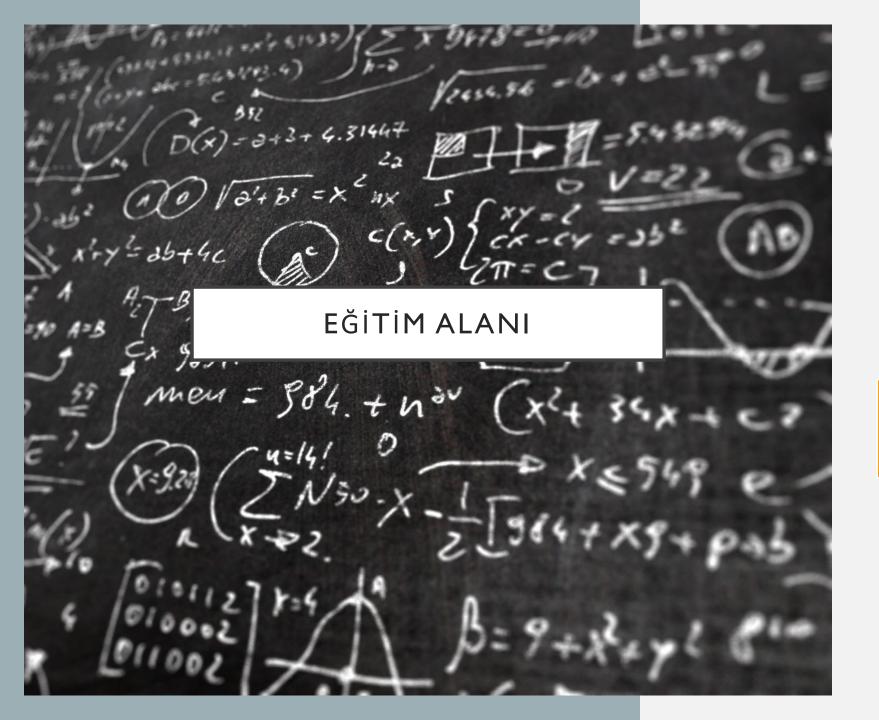
E-Ticaret Siteleri



Hastane Sistemleri



Devlet Siteleri



Ders Planları Ders Modülleri

Ödevler

Dil Eğitimi



Telegram gibi sosyal medya aygıtlarının ve Chatbotların yabancı dil eğitiminde kullanılması ile ilgili araştırmalar mevcuttur.





Ismail Xodabande

Kharazmi Üniversitesi – Yabancı Diller Bölümü

Sosyal medyanın İngilizce telaffuzundaki etkisini Telegram arayıcılığıyla test etmek istenen araştırmada ortalama İngilizce düzeyleri B1 olan 30 İranlı EFL (Yabancı dil olarak İngilizce) öğrencisi 14 kişilik deney ve 16 kişilik kontrol grubu olmak üzere iki gruba ayrılmıştır. Kontrol grubu İngilizce eğitimlerine öğretmenler aracılığıyla devam ederken, deney grubuna Telegram üzerinden günde 5 kere olmak koşuluyla İngilizce bir kelime ve doğru telaffuzu paylaşılmıştır. 4 Hafta süren bu deneyde Telegram üzerinden doğru telaffuzları alan deney grubu öğrencilerinin sınavlarında daha başarılı olduğu görülmüştür. Bu çalışma yabancı dil eğitiminde sosyal medyanın kullanımının etkili ve umut verici olduğunu ortaya çıkarmıştır



Dr. Jiyou Jia Pekin Üniversitesi- Eğitim Teknolojileri Bölümü

Web tabanlı Chatbot sisteminin yabancı dil öğrenimi ile ilgili yapılan çalışmada sistem Çin'de bulunan bir sunucuya kurulmuştur. Sistemin amacı kullanıcıların İngilizce ya da Almanca dillerinde bot ile karşılıklı sohbet edebilmesidir. Uygulamaya kullanıcıların bilgi alabileceği temel bir Çince de eklenmiş. Kullanıcıların çoğu Çin'de bulunan ve İngilizce dilini akıcı bir şekilde yazıp konuşabilen üniversite öğrencileridir. Öğrenciler bu uygulamadan bir ilan tahtasında verilen reklam aracılığıyla bilgilendirilmiştir. Kullanıcıların siteye girip botu kullanma sıklığı, bot ile konuştukları konular, ve bota yapılan yorumların değerleri çıkarılmıştır. Çalışmanın sonucunda eğer yabancı dil öğretimi için bir chatbot sistemi başarıyla uygulanabilirse bu konuda devrim niteliğinde bir kilometre taşı olacağı saptanmıştır. Bu sistemin öğretmenlerin yükünü hafifletebileceği ve öğrencilere istedikleri zaman yabancı dillerini geliştirebilecekleri bir ortam sağlayabileceği belirlenmiştir.



Hsiu-Ling Chen
Ulusal Taiwan Üniversitesi – Bilim Ve Teknoloji Bölümü

Bir Chatbot uygulamasının Çince öğrenme üzerindeki etkilerini Teknoloji kabul modelini referans alarak inceleyen çalışmada Taiwanda bulunan bir üniversitede farklı milletlerden katılımcılar bulunduran 58 kişilik bir grup kullanılmıştır. Katılımcılar 39 kişilik kontrol grubu ve 19 kişilik deney grubu olmak üzere iki farklı gruba ayrılmıştır. Kontrol grubu Chatbotu bire çoklu öğrenim metoduyla, deney grubu ise bire bir öğrenim metoduyla kullanmıştır. Veri analizleri sonucu Chatbot'un öğrencilerin öğrenme başarısını önemli derecede arttırdığı ve bire bir yöntem kullanılarak yapılan çalışmanın sınıf ortamında alınan dersten daha iyi bir sonuç verebileceği gözlemlenmiştir. Teste katılan öğrenciler Chatbot'un Çince öğrenmelerine katkıda bulunduğunu kabul ettiler.



YAPILAN PROJENIN TANITIMI



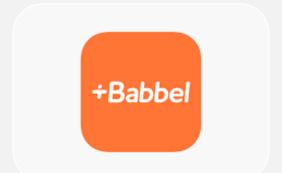




Duolingo

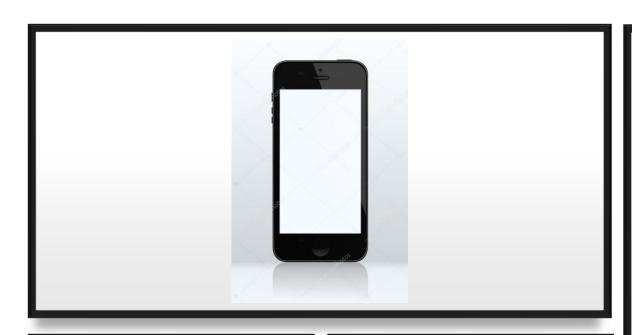
Live Mocha

Busuu



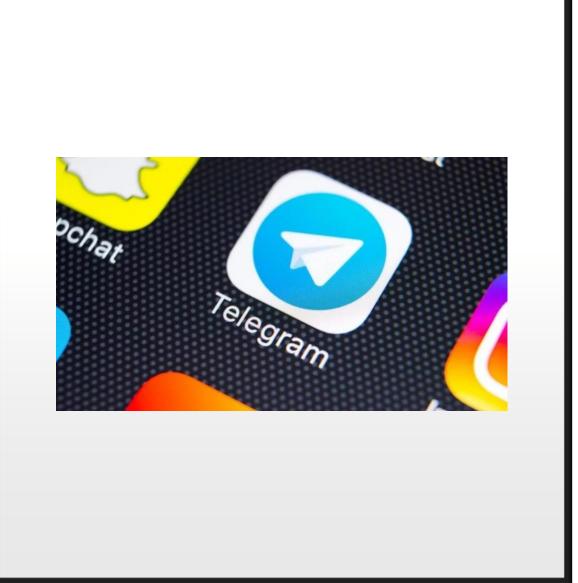
Babbel











PROJEDE KULLANILAN TEKNOLOJILER







Node JS

PROJEDEN KOD ÖRNEKLERI



EXPRESS SERVER, DATABASE VE ROUTER TANIMLAMASI

```
const express = require("express");
const connectDataBase = require("./helpers/connectDataBase");
const dotenv =require("dotenv");
const router = require("./routers/index");
dotenv.config({
    path:"./config/env/config.env"
});
const PORT = 5000 || process.env.PORT;
connectDataBase();
const app = express();
app.use(express.json());
app.use("/telegramApi" ,router);
                                              18/26
app.listen(PORT , () => {
    console.log(`Server active and running on port : ${PORT}`);
})
```

USER VE SORU MODELLERI

```
const Schema = mongoose.Schema;
const User = new Schema({
       type:String,
       required:true,
       unique:true
   first_name:{
        type:String
   last_name:{
       type:String
   current_question_english:{
       type:Number,
       required:true,
        default:0
   current_question_german:{
       type:Number,
       required:true,
        default:0
   current_question_french:{
       type:Number,
       required:true,
       default:0
   current_question_russian:{
        type:Number,
       required:true,
        default:0
module.exports = mongoose.model("User", User);
```

```
const mongoose= require("mongoose");
const Schema = mongoose.Schema;
const QuestionEnglish = new Schema({
   question : {
       type:String,
       required:true,
       unique:true
   answer:{
       type:Array,
       required:true
   lang:{
       type:String,
       default: "English"
   explanation:{
       type:String,
       default: "No Explanation Avaible For This Question"
                                                                   19/26
module.exports = mongoose.model("EnglishQuestion" , QuestionEnglish);
```

ÖRNEK CONTROLLER KODU

```
const postQuestionSingleEnglish = async (req , res , next) => {
   try{
       const info = req.body;
       const question = new QuestionEnglish({
            ...info
       await question.save();
       res.status(201).json({
           question
   }catch(err){
       res.json({
           err:err
const postQuestionManyEnglish = async (req , res, next) => {
   try{
       const questions= await QuestionEnglish.create(ydsSorularEnglish);
       res.status(201).json({
           questions
   }catch(err){
       res.json({
           err:err.message
const getQuestionEnglish = async(req, res ,next) => {
   try{
       const questions = await QuestionEnglish.find();
       res.status(200).json(questions);
   }catch(err){
       res.status(404).json({
                                                              20/26
           err:err
```

TELEGRAF ILE BOT TANIMLAMASI

```
const bot = new Telegraf(key);
let encounter = 0:
let gercounter = 0;
let frcounter = 0;
let rucounter=0;
const mainMenuKeyboard = (new Keyboard())
    .add('Test Seç','Yardım')
   bot.start((ctx) => {
       const info = ctx.message.chat;
       ctx.reply(`Hoş Geldin ${info.first_name} ${info.last_name}, Bu bot İngilizce, Almanca, Fransızca ve Rusça dillerindeki bilgilerinizi test etmek için geliştirilmiştir
       postUser(info.id,info.first_name,info.last_name)
       .catch((er) => {
           console.log(er);
    .hears('Yardım', (ctx) => {
       ctx.reply('Yardım texti', mainMenuKeyboard.draw())
   .hears('Test Seç', ctx => {
       const keyboard = (new Keyboard())
       .add('YDS İngilizce', 'YDS Almanca')
       .add('YDS Rusça' , 'YDS Fransızca')
       ctx.reply('Test Seciniz', keyboard.draw())
```

UYGULAMA İÇERİĞİ

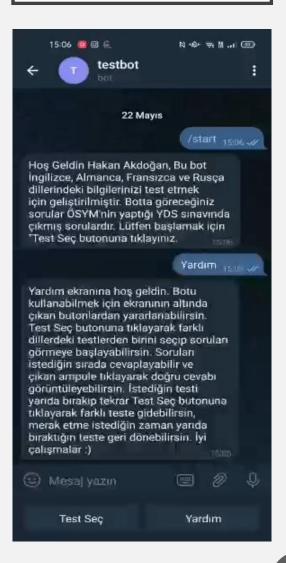
HOŞGELDINIZ EKRANI



YARDIM EKRANI



TEST SEÇİMİ EKRANI



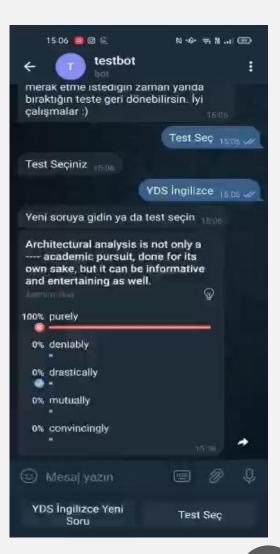
SORU EKRANI



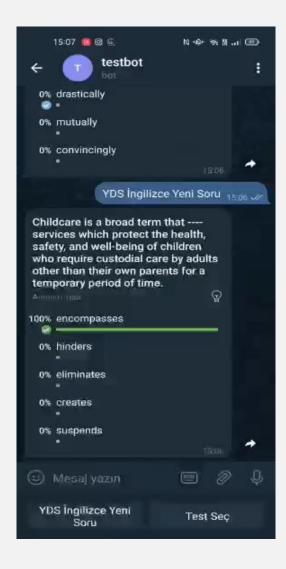
YANLIŞ CEVAP EKRANI



DOĞRU CEVAP EKRANI



AÇIKLAMA EKRANI



TEŞEKKÜRLER!

https://github.com/hakanakdogan/hakanakdogan-tubitak-proje

t.me/hakanTestBot_bot