



**T.C.
AKDENİZ ÜNİVERSİTESİ**

Elektrik-Elektronik Mühendisliği Bölümü

Bitirme Çalışması

**Kripto Portföy Yöneticisi: Arduino Devreli Telegram Botu ile
Akıllı Yatırım**

Ahmad AGHAKISHIYEV

Danışman: Dr. Öğr. Üyesi Yalçın ALBAYRAK

HAZİRAN 2025

Antalya

Bu çalışma / / 2025 tarihinde aşağıdaki jüri tarafından Elektrik-Elektronik Mühendisliği Bölümü'nde Lisans Bitirme Projesi olarak kabul edilmiştir.

Bitirme Projesi Jürisi

| | | |
|--------------|-----------------------|--|
| Danışman Adı | | |
| Üniversite | Akdeniz Üniversitesi | |
| Fakülte | Mühendislik Fakültesi | |

| | | |
|------------|-----------------------|--|
| Jüri Üyesi | | |
| Üniversite | Akdeniz Üniversitesi | |
| Fakülte | Mühendislik Fakültesi | |

| | | |
|------------|-----------------------|--|
| Jüri Üyesi | | |
| Üniversite | Akdeniz Üniversitesi | |
| Fakülte | Mühendislik Fakültesi | |

| | | |
|--------------|-----------------------|--|
| Danışman Adı | | |
| Üniversite | Akdeniz Üniversitesi | |
| Fakülte | Mühendislik Fakültesi | |

| | | |
|------------|-----------------------|--|
| Jüri Üyesi | | |
| Üniversite | Akdeniz Üniversitesi | |
| Fakülte | Mühendislik Fakültesi | |

| | | |
|------------|-----------------------|--|
| Jüri Üyesi | | |
| Üniversite | Akdeniz Üniversitesi | |
| Fakülte | Mühendislik Fakültesi | |

ÖZET

Bu çalışma, kripto para piyasasında bireysel yatırımcıların ihtiyaç duyduğu hızlı, güvenilir ve kullanıcı dostu bir yardımcının oluşturulması amacıyla geliştirilmiştir. Geliştirilen sistem; Telegram Bot API, CoinGecko API, Python, SQLite ve Arduino teknolojilerini bir araya getirerek, kullanıcıların anlık fiyat takibi yapabilmesini, geçmiş verilere dayalı grafik oluşturmasını, yatırım listesini yönetmesini ve otomatik uyarılar almasını sağlamaktadır. Kullanıcılar, Telegram üzerinden botla etkileşime geçerek coin fiyatlarını öğrenebilir, kişisel portföy oluşturabilir ve yatırım durumlarına göre hem metinsel hem de fiziksel bildirim alabilirler. Sistemin Arduino bileşeni sayesinde, yatırımın kazançlı ya da zararda olması durumuna göre LED ve sesli bildirimle fiziksel bir uyarı sağlanmaktadır.

Bu proje, kripto para piyasasında veri temelli yatırım kararlarının alınmasını teşvik ederek, kullanıcıların duygusal kararlardan uzak durmasını amaçlamaktadır. Veriler, kullanıcı gizliliğine önem verilerek SQLite veritabanında şifreli bir şekilde saklanmıştır. Aynı zamanda sistem; maliyet açısından düşük, kullanımı kolay ve tamamen açık kaynak bir çözüm sunmaktadır. Sonuç olarak, bireysel yatırımcıların piyasaya daha bilinçli yaklaşmalarına yardımcı olacak ve kripto varlık yönetimini daha erişilebilir hale getirecek bütüncül bir çözüm geliştirilmiştir.

Anahtar Kelimeler:API,Arduino,bot.coin,coingecko,grafik,kripto,telegram

ÖNSÖZ

Bu çalışmanın hayata geçirilmesinde emeği geçen herkese en içten teşekkürlerimi sunarım.

Özellikle, eğitim hayatım boyunca bana yol gösteren, bilgi ve tecrübesiyle her zaman destek olan değerli danışman hocam Yalçın ALBAYRAK'a teşekkür ederim. Onun rehberliği olmasaydı, bu projeyi bu düzeye taşıyamazdım.

Gerek moral, gerek teknik destek konusunda yanımda olan değerli arkadaşlarıma, kodlama aşamalarında fikir alışverişinde bulunduğum sınıf arkadaşlarıma ve proje sürecinde karşılaştığım zorluklarda bana destek olan herkese ayrı ayrı minnettarım.

Saygılarımla,

Ahmad AGHAKISHIYEV

İÇİNDEKİLER

| | |
|--|----|
| ÖZET | 4 |
| ÖNSÖZ | 5 |
| İÇİNDEKİLER | 6 |
| SİMGELER VE KISALTMALAR | 8 |
| ŞEKİLLER DİZİNİ | 9 |
| TABLolar DİZİNİ | 10 |
| Tablo 5.2. Performans Analiz | |
| Tablosu | 25 |
| 1.GİRİŞ | 11 |
| 1.1. Problem Tanımı | 11 |
| 1.2. Çalışmanın amacı | 12 |
| 1.3. Çalışmanın Önemi | 12 |
| 1.4. Kapsam ve Sınırlılıklar | 12 |
| 2.LİTERATÜR TARAMASI | 13 |
| 2.1. Kripto Para Piyasası ve Yatırımcı Davranışları | 13 |
| 2.2. Telegram Botları ve Finansal Uygulamalardaki Rolü | 13 |
| 2.3. CoinGecko API ve Kripto Piyasası Veri Kaynakları | 14 |
| 2.4. Arduino ile Fiziksel Bildirim Sistemleri | 14 |
| 2.5. Veri Güvenliği ve SQLite Entegrasyonu | 14 |
| 3.SİSTEM TASARIMI VE MİMARİ YAPI | 15 |
| 3.1 Sistem Genel Mimarisi | 15 |
| 3.2 Telegram Bot Mimarisi | 15 |
| 3.3 Veri Akışı Diyagramı | 16 |
| 3.4 Donanım-Yazılım Entegrasyonu | 17 |
| 4.GELİŞTİRME ORTAMI VE UYGULAMA DETAYLARI | 19 |
| 4.1 Yazılım Geliştirme Ortamı ve Kullanılan Teknolojiler | 19 |
| 4.2 Telegram Bot Komut Akışı | 20 |
| 4.3 Veritabanı ve Veri Modeli | 20 |

| | |
|---|----|
| 4.4 Gerçek Zamanlı Yatırım Güncelleme _____ | 21 |
| 4.5 Grafik Oluşturma ve Gönderme _____ | 21 |
| 4.6 Hata Yönetimi ve Loglama _____ | 22 |
| 4.7 Arduino Yazılım Mimarisi _____ | 22 |
| 4.8 Donanım Kaynak Yönetimi ve Elektriksel Özellikler _____ | 22 |
| 5.SİSTEM TESTLERİ VE DEĞERLENDİRME _____ | 23 |
| 5.1 Test Ortamı ve Hazırlık Süreci _____ | 23 |
| 5.2 Test Edilen Senaryolar ve Akışlar _____ | 23 |
| 5.3 Karşılaşılan Sorunlar ve Çözüm Önerileri _____ | 24 |
| 5.3.1 API Erişim Problemleri _____ | 24 |
| 5.3.2 Telegram Bot Gecikmeleri _____ | 24 |
| 5.3.3 Talkie Ses Sorunu _____ | 24 |
| Tablo 5.2. Performans Analiz Tablosu _____ | 24 |
| 5.4 Genel Değerlendirme _____ | 24 |
| 6.SONUŞLAR VE ÖNERİLER _____ | 25 |
| 6.1 Sistemsel Kazanımlar _____ | 25 |
| Tablo 6.1 Teknik Başarı tablosu _____ | 25 |
| 6.3 Toplumsal Katkı ve Kullanım Alanları _____ | 26 |
| 6.4 Geliştirme Önerileri _____ | 26 |
| 6.5 Sonuç _____ | 27 |
| 7.KAYNAKLAR _____ | 28 |
| Özgeçmiş _____ | 29 |

SİMGELER VE KISALTMALAR

1. API (Application Programming Interface): Yazılımların birbirleriyle iletişim kurmasını sağlayan arayüz (CoinGecko API, Telegram Bot API).
2. SQLite: Hafif, sunucu gerektirmeyen bir veritabanı yönetim sistemi.
3. LED (Light Emitting Diode): Arduino üzerinde görsel uyarı için kullanılan ışık yayan diyot.
4. JSON (JavaScript Object Notation): CoinGecko API'den alınan verilerin formatı.
5. CRUD (Create, Read, Update, Delete): Veritabanı işlemlerinin temel fonksiyonları.
6. FOMO (Fear of Missing Out): Kripto piyasasında yaygın bir psikolojik davranış biçimi.

ŞEKİLLER DİZİNİ

| | |
|--|----|
| Şekil 3.1. Veri-akış diyagramı | 18 |
| Şekil 3.2. Arduino devresi üstten görüntüsü. | 19 |
| Şekil 4.1. Telegram Grafik ekran görüntüsü. | 22 |
| Şekil 4.2. Try-Except yapı kullanımı. | 23 |

TABLOLAR DİZİNİ

| | |
|--|----|
| Tablo 3.1.Komutlar | 16 |
| Tablo 3.2.Donanım tablosu | 18 |
| Tablo 3.3.Arduino pin konfigürasyonu | 18 |
| Tablo 4.1.Geliştirme ortamı ve Uygulama detayları. | 20 |
| Tablo 4.2.Modül Tablosu | 20 |
| Tablo 4.3.Veritabanı tablosu 1. | 21 |
| Tablo 4.4.Veritabanı tablosu 2. | 21 |
| Tablo 4.5. Donanım tablosu | 23 |
| Tablo 5.1.Test Tablosu | 24 |
| Tablo 5.2. Performans Analiz Tablosu | 25 |
| Tablo 6.1 Teknik Başarı tablosu | 26 |

1.GİRİŞ

Kripto paralar, dijital finansal ekosistemin son yıllardaki en dinamik ve dönüştürücü varlık sınıfı olarak öne çıkmaktadır. Merkeziyetsiz yapıları, küresel erişim imkanı ve yüksek kazanç özellikleriyle bu piyasalar, yatırımcılar için önemli fırsatlar sunarken aynı zamanda ciddi riskler de barındırmaktadır. Özellikle bireysel yatırımcılar, piyasanın sürekli değişen dinamiklerini takip etme zorluğu ve duygusal karar verme eğilimleri nedeniyle beklenmedik finansal kayıplarla karşılaşa bilmektedir.

Bu tez çalışması, bireysel yatırımcıların kripto para piyasalarında veriye dayalı, zamanında ve rasyonel kararlar alabilmesini sağlamak amacıyla, Telegram tabanlı entegre bir kripto portföy yönetim ve bildirim sistemi geliştirmeyi hedeflemektedir. Tasarlanan sistem, Python programlama dili kullanılarak geliştirilen bir Telegram botu aracılığıyla kullanıcıların;

- Gerçek zamanlı kripto para fiyatlarını sorgulamasına,
- Portföy izlemesine,
- Fiyat alarmları almasına,
- Tarihsel veri analizi ve grafiksel görselleştirmelere erişmesine olanak tanımaktadır.

Bunun yanı sıra, sistemin yenilikçi bir bileşeni olarak Arduino tabanlı bir fiziksel bildirim mekanizması entegre edilmiştir. Bu mekanizma, LED aydınlatma ve sesli uyarılar aracılığıyla kullanıcıları kritik piyasa hareketlerine karşı çoklu-duyusal bir deneyimle bilgilendirmektedir.

Bu giriş bölümünde, çalışmanın temel problemi ve motivasyonu ortaya konacak, araştırmanın amacı ve önemi detaylandırılacak, ayrıca çalışmanın kapsamı ve sınırlılıkları açıklanacaktır.

1.1. Problem Tanımı

Kripto para piyasaları 7/24 kesintisiz işlem özellikleriyle geleneksel finansal piyasalardan ayrılmaktadır. Bu durum, yatırımcıların piyasa hareketlerini sürekli takip etmesini gerektirmekte ve özellikle bireysel yatırımcılar için aşağıdaki zorlukları beraberinde getirmektedir:

- Anlık fiyat değişimlerinin manuel takibinin zorluğu,
- Karmaşık ticaret platformlarındaki arayüzlerin kullanıcı dostu olmaması,
- Duygusal karar verme eğiliminin yatırım performansını olumsuz etkilemesi,
- Çoklu veri kaynakları arasında bilgi kirliliği ve analiz zorluğu.

Mevcut çözümlerin bir kısmı, teknik analiz araçları sunsa da, çoğunlukla karmaşık arayüzler içermekte ve mobil kullanıcılar için pratik erişim sağlamamaktadır. Bu bağlamda, basit, erişilebilir ve çok kanallı bir bildirim sistemi, yatırımcıların karar verme süreçlerini optimize edebilir.

1.2. Çalışmanın amacı

Bu çalışmanın temel amacı, kripto para yatırımcılarının: 1.Telegram üzerinden kolay ve hızlı bir şekilde piyasa verilerine erişmesini, 2.Kişiselleştirilmiş portföy takibi yapabilmesini, 3.Otomatik bildirimlerle kritik fiyat hareketlerinden anında haberdar olmasını, 4.Arduino tabanlı fiziksel uyarı sistemleriyle çoklu-duyusal geri bildirim almasını sağlayan entegre bir yazılım ve donanım çözümü geliştirmektir.

Bu sistem, kullanıcıların rasyonel ve veriye dayalı kararlar almasına yardımcı olarak, kripto para yatırımlarındaki riskleri minimize etmeyi hedeflemektedir.

1.3. Çalışmanın Önemi

Kripto para piyasalarındaki saniyelik fiyat dalgalanmaları, yatırımcılar için hem fırsat hem de tehdit oluşturmaktadır. Bu çalışma, aşağıdaki nedenlerle önem taşımaktadır:

1. Zamanında ve doğru bilgiye erişim, yatırımcıların stratejik kararlar almasını kolaylaştırır.
2. Telegram botu entegrasyonu, kullanıcıların herhangi bir ekstra uygulama yüklemeden anlık veri almasını sağlar.
3. Arduino ile fiziksel bildirimler, görsel ve işitsel uyarılar sayesinde kullanıcıların dikkatini çekerek tepki süresini kısaltır.
4. Bireysel yatırımcıların finansal okuryazarlığını artırarak, spekülasyon hareketlerin önüne geçilmesine katkı sağlar.

1.4. Kapsam ve Sınırlılıklar

Kapsam:

- Sistem, Python tabanlı bir Telegram botu üzerinden metin tabanlı komutlarla çalışır.
- CoinGecko API kullanılarak gerçek zamanlı ve tarihsel kripto para verileri sağlanır.
- Kullanıcılar /fiyat, /yatırımlarım, /grafik gibi komutlarla etkileşim kurar.
- Arduino UNO ve ilgili sensörler (LED, hoparlör) ile fiziksel bildirim sistemi entegre edilmiştir.

Sınırlılıklar:

- API Kısıtlamaları: CoinGecko API'nin ücretsiz sürümündeki istek limitleri, sistem performansını etkileyebilir.
- Platform Bağımlılığı: Şu anlık yalnızca Telegram üzerinden erişim sağlanmaktadır.
- Donanım Kısıtları: Arduino bildirim sistemi, yalnızca temel LED ve ses uyarılarıyla sınırlıdır.

2.LİTERATÜR TARAMASI

Bu bölümde, kripto para piyasalarının yapısı, Telegram botlarının finansal sistemlerdeki rolü, Arduino tabanlı bildirim mekanizmaları ve kullanıcı odaklı portföy yönetim uygulamalarına ilişkin mevcut literatür incelenmiştir. Ayrıca, bu çalışmanın temel bileşenleri olan Telegram Bot API, CoinGecko API ve Arduino entegrasyonu hakkında detaylı bir değerlendirme yapılmıştır.

2.1. Kripto Para Piyasası ve Yatırımcı Davranışları

Kripto para piyasaları, merkeziyetsiz yapıları ve blockchain teknolojisiyle desteklenen güvenli işlem altyapıları sayesinde geleneksel finansal sistemlerden ayrılmaktadır. Bitcoin'in (BTC) 2009 yılında ortaya çıkışından bu yana, binlerce alternatif kripto para birimi (altcoin) piyasaya girmiş ve küresel bir dijital varlık ekosistemi oluşmuştur.

Literatürde, kripto para yatırımcılarının davranışlarını inceleyen çalışmalar, bu piyasanın yüksek volatiliteye sahip olması nedeniyle yatırımcıların genellikle duygusal kararlar aldığını göstermektedir. Özellikle:

- FOMO (Fear of Missing Out – Kaçırma Korkusu): Yükselen piyasalarda kontrolsüz alım yapma eğilimi,
- Panik Satış (Capitulation): Ani fiyat düşüşlerinde mantıksız satışlar yapma,
- Aşırı Güven (Overconfidence): Teknik analiz yapmadan spekülasyon yatırımlara yönelme gibi psikolojik faktörler, yatırımcıların performansını olumsuz etkilemektedir.

Bu nedenle, veriye dayalı ve otomatik karar destek sistemleri, yatırımcıların rasyonel hareket etmesine yardımcı olabilir.[8]

2.2. Telegram Botları ve Finansal Uygulamalardaki Rolü

Telegram botları, kullanıcıların metin tabanlı komutlarla etkileşime girebildiği otomatik yazılımlardır. Özellikle finansal uygulamalarda, anlık fiyat sorgulama, portföy takibi, özelleştirilmiş alarmlar, haber ve analiz bildirimleri gibi işlevler sunmaktadır.

Botfather ile Telegram Bot Geliştirme:

Botfather, Telegram'ın resmi bot yönetim aracıdır ve geliştiricilere yeni botlar oluşturma, API anahtarı edinme ve bot ayarlarını yapılandırma imkanı tanır. Çalışma prensibi şu şekildedir:

- Bot Oluşturma: /newbot komutu ile yeni bir bot tanımlanır ve benzersiz bir API token alınır.
- Komut Yapılandırması: /setcommands ile botun desteklediği komutlar (örneğin /coinlerim, /coin_ekle) tanımlanır veya Python kodu ile özel fonksiyonlar oluşturularak yeni komutlar yaratılır.
- Webhook veya Polling Yöntemi: Bot, ya Telegram sunucularına sürekli istek atarak (Polling) ya da belirli bir sunucuya veri göndererek (Webhook) çalışır.

Bu çalışmada, Python ve python-telegram-bot kütüphanesi kullanılarak, kullanıcı dostu bir portföy yönetim botu geliştirilmiştir[2].

2.3. CoinGecko API ve Kripto Piyasası Veri Kaynakları

Kripto para fiyatları, piyasa hacmi ve tarihsel veriler gibi bilgileri çekmek için CoinGecko API kullanılmıştır. Bu API'nin avantajları şunlardır[1]:

1. Ücretsiz ve geniş kapsamlı veri erişimi,
2. JSON formatında kolay işlenebilir yanıtlar,
3. Binlerce kripto para birimini desteklemesi,
4. Gerçek zamanlı ve tarihsel veri sağlama yeteneği.

Ancak, ücretsiz sürümde dakikada 10-50 istek limiti bulunması, yüksek trafikli durumlarda sistem performansını etkileyebilir.

2.4. Arduino ile Fiziksel Bildirim Sistemleri

Donanımsal uyarı mekanizmaları, kullanıcıların yalnızca dijital bildirimlere bağımlı kalmadan fiziksel geri bildirim almasını sağlar. Arduino tabanlı sistemlerde:

- 1.LED'ler (örneğin yeşil LED = fiyat artışı, kırmızı LED = düşüş),
- 2.Hoparlör (sesli uyarı) ile kritik seviyelerde alarm,

Çoklu duyuşsal uyarı sistemleri, kullanıcıların dikkatini daha etkili bir şekilde çekmektedir.

2.5. Veri Güvenliği ve SQLite Entegrasyonu

Kullanıcı portföy bilgileri gibi hassas verilerin güvenliği için SQLite veritabanı kullanılmıştır. SQLite'ın avantajları arasında, hafif ve hızlı bir veritabanı yönetim sistemi, şifreleme desteği ve sunucu gerektirmeyen yerel depolama özellikleri bulunmaktadır[3].

3.SİSTEM TASARIMI VE MİMARİ YAPI

Bu bölümde, geliştirilen Telegram bot tabanlı kripto portföy yöneticisinin mimari bileşenleri, veri akışı ve donanım-yazılım entegrasyonuna dair ayrıntılar ele alınacaktır. Her alt başlık, insan odaklı bir anlatımla, okuyucunun sistemi bütünsel olarak kavramasını sağlayacak şekilde formüle edilmiştir.

3.1 Sistem Genel Mimarisi

Günümüz finansal teknolojilerinde hız ve güvenlik en kritik iki faktördür. Bu tezde önerilen sistem de bu ihtiyaçlar doğrultusunda üç ana bileşeni bir araya getirir. Öncelikle, kullanıcılar Telegram Bot Arayüzü üzerinden komut göndererek etkileşime geçer. Ardından, komutlar Python Sunucu Kod Yapısı içinde işlenir; burada CoinGecko API'den alınan fiyat verileri SQLite veritabanına kaydedilir ve analiz edilir. Son olarak, sistemin fiziksel ayağını oluşturan Arduino Donanım Modülü, Python'dan gelen “kazanç” ya da “zarar” sinyallerini LED ve hoparlör aracılığıyla kullanıcıya iletir. Bu üç katmanlı mimari, hem kullanıcı dostu bir deneyim hem de yüksek performans sunar.

3.2 Telegram Bot Mimarisi

Telegram Bot API'nin esnek yapısı sayesinde, botumuz asenkron bir şekilde komutları alır ve karşılık verir. Kullanıcılar, sohbet ekranında aşağıdaki komutları mesai gözetmeksizin kullanabilir. Her komut, Python modülünde ilgili fonksiyona yönlendirilir ve sonuçlar anında kullanıcıya iletir.

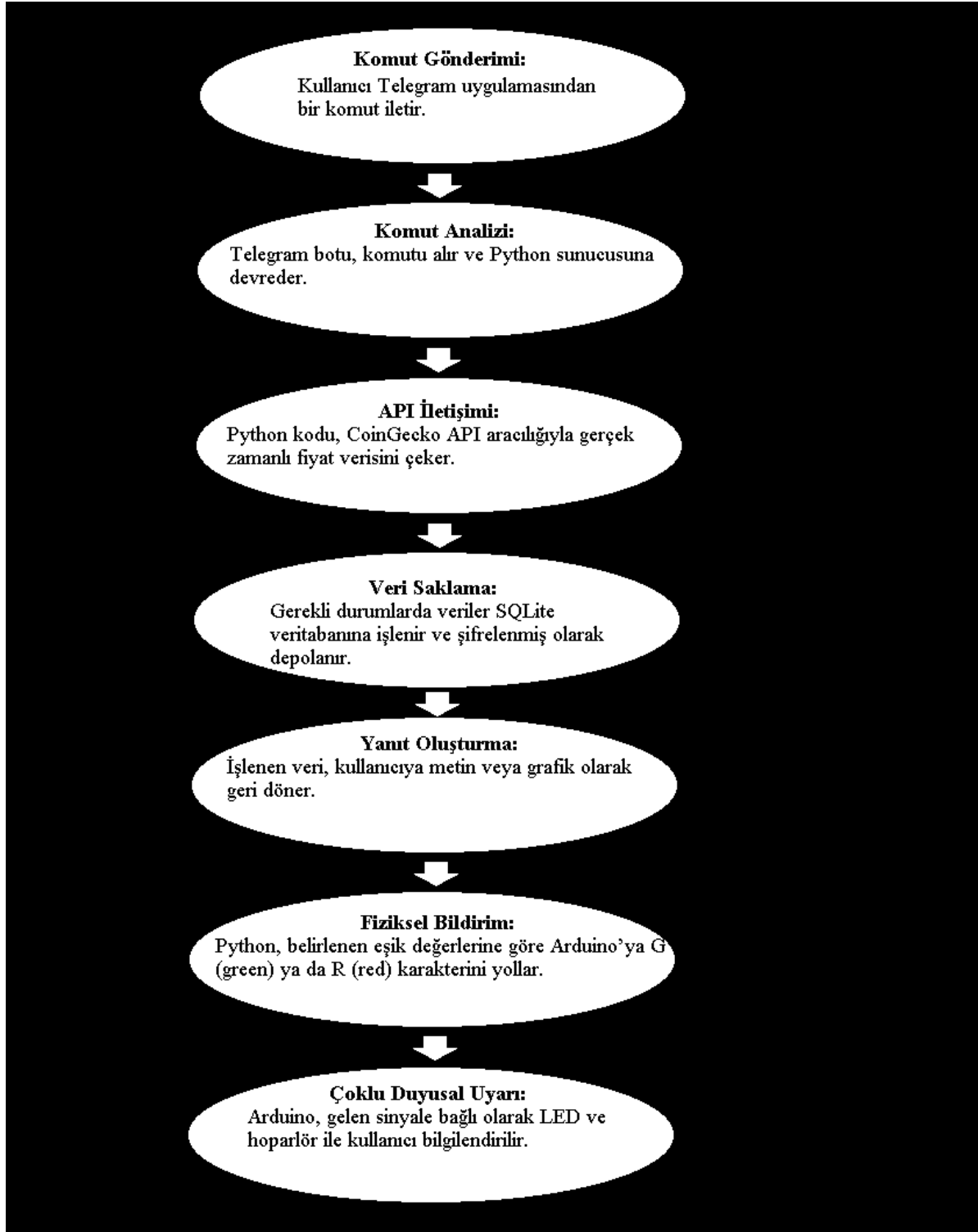
TABLO 3.1.Komutlar

| Komut | Parametreler | Açıklama |
|---------------|----------------------|-------------------------------|
| /baslat | - | Başlangıç menüsünü gösterir |
| /coin_ekle | coin_ismi | Portföye yeni coin ekler |
| /yatirim_yap | coin_ismi miktar | Yatırım kaydı oluşturur |
| /grafik | coin_ismi gün_sayısı | Fiyat grafiği gösterir |
| /coinlerim | - | Portföydeki coinleri listeler |
| /yatirimlarim | - | Yatırım geçmişini gösterir |
| /coin_sil | coin_ismi | Coin'i portföyden çıkarır |

3.3 Veri Akışı Diyagramı

Veri yolculuğu, kullanıcının Telegram üzerinden komut göndermesiyle başlar ve fiziksel uyarı sisteminin tetiklenmesiyle son bulur. Aşağıdaki adımlar, bu akışı anlaşılır bir dilde özetlemektedir:

Şekil 3.1. Veri-akış diyagramı.



3.4 Donanım-Yazılım Entegrasyonu

Donanım tarafında Arduino Uno SMD kullanılmıştır. Python sunucusu, USB seri port üzerinden geleneksel seri iletişim kanalı ile tek karakterlik komut ('G' veya 'R') göndererek donanımı yönlendirir. Kablo üzerinden Arduino'ya uygulanan elektriksel besleme özellikleri aşağıdaki gibidir:

Tablo 3.2.Donanım tablosu

| Bileşen | Değer |
|-----------------------|-------------------------|
| Besleme gerilimi(USB) | 5 V DC $\pm 5\%$ |
| Maks. Çalışma Akımı | 500 mA(USB port limit) |
| Çalışma Frekansı | 490 Hz |

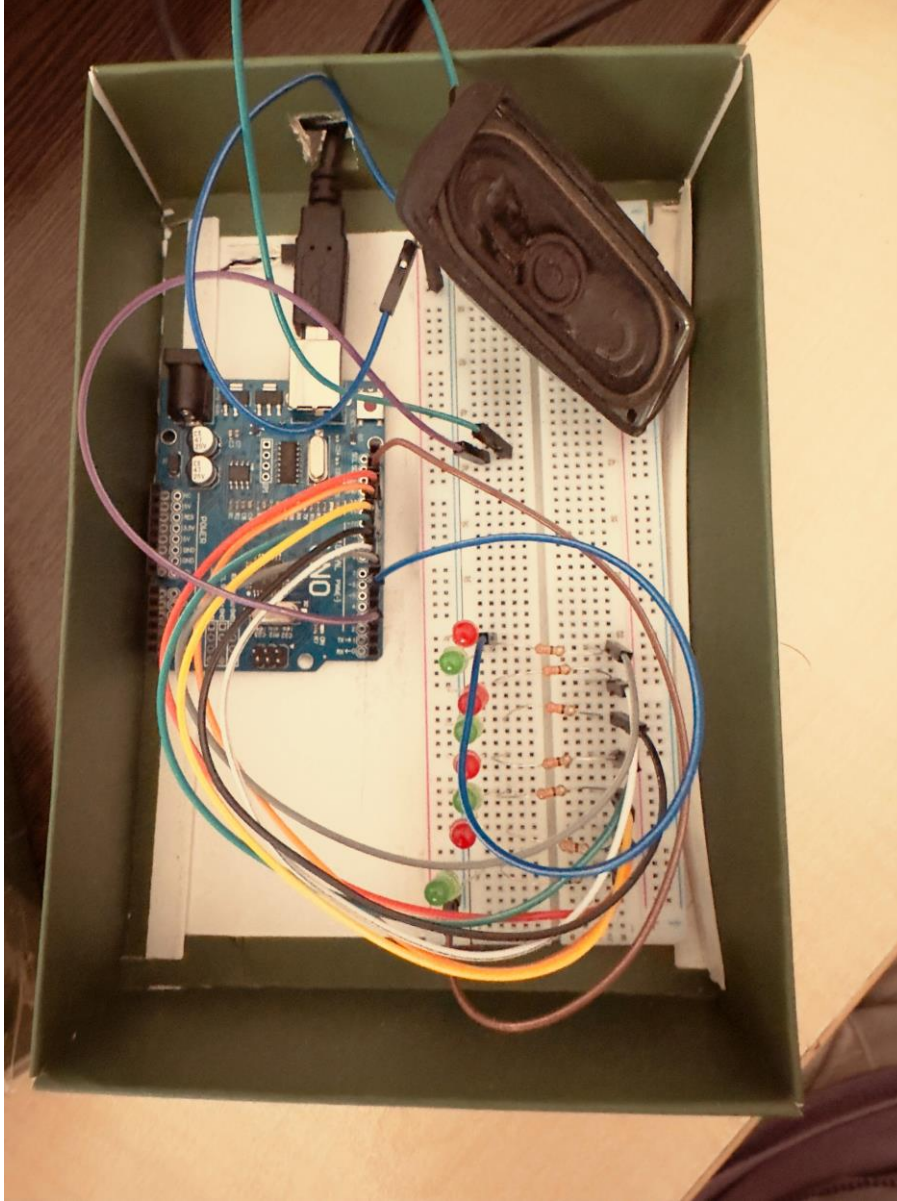
Arduino üzerindeki pin konfigürasyonu ise şu şekildedir:

Tablo 3.3.Arduino pin konfigürasyonu

| İşlev / LED | Arduino Pin No | Notlar |
|---------------|----------------|-------------------------------|
| Yeşil LED 1 | D13 | Harici, 220 Ω dirençli |
| Yeşil LED 2 | D11 | Harici, 220 Ω dirençli |
| Yeşil LED 3 | D9 | Harici, 220 Ω dirençli |
| Yeşil LED 4 | D7 | Harici, 220 Ω dirençli |
| Kırmızı LED 1 | D12 | Harici, 220 Ω dirençli |
| Kırmızı LED 2 | D10 | Harici, 220 Ω dirençli |
| Kırmızı LED 3 | D8 | Harici, 220 Ω dirençli |
| Kırmızı LED 4 | D6 | Harici, 220 Ω dirençli |
| Hoparlör | D3 (PWM) | 490hz civarı ses üretir |

Bu konfigürasyon ile seri port üzerinden gelen pozitif sinyalde yeşil LED'ler, negatif sinyalde kırmızı LED'ler ve hoparlör üzerinden farklı frekanslarda sesli geri bildirim sağlanmaktadır.

Şekil 3.2.Arduino devresi üstten görüntüsü.



4.GELİŞTİRME ORTAMI VE UYGULAMA DETAYLARI

Bu bölümde, Telegram tabanlı kripto portföy yönetim sisteminin geliştirilmesinde kullanılan yazılım ve donanım araçları ayrıntılı olarak ele alınmakta, kullanılan teknolojiler, yazılım geliştirme dili ve geliştirme süreçleri açıklanmaktadır. Ayrıca Arduino UNO SMD tipi donanım bileşeniyle gerçekleştirilen yazılım entegrasyonu, kullanılan pin tanımlamaları ve enerji iletim detayları da kapsamlı bir biçimde sunulmaktadır .

4.1 Yazılım Geliştirme Ortamı ve Kullanılan Teknolojiler

Projenin yazılım kısmı Python dili kullanılarak geliştirilmiştir. Python, özellikle hızlı prototipleme ve API entegrasyonu için tercih edilmiştir. Bot yapısında python-telegram-bot kütüphanesi kullanılmış, kripto para verileri için CoinGecko API entegre edilmiştir. Veritabanı yönetimi için gömülü ve hafif bir veritabanı olan SQLite tercih edilmiştir.

Tablo 4.1.Geliştirme ortamı ve Uygulama detayları.

| Bileşen | Açıklama |
|---------------------|-----------------------------------|
| Python 3.10 | Geliştirme dili |
| python-telegram-bot | Telegram API ile bot entegrasyonu |
| requests | API istekleri için |
| matplotlib | Coin grafiklerinin çizimi |
| SQLite3 | Veritabanı yönetimi |
| Arduino IDE | Arduino Uno programlaması |

Proje, Telegram bot işlevleri, veri erişim katmanı, API entegrasyonu ve seri port haberleşmesi olmak üzere dört ana modülden oluşmaktadır. Modüler yapı, bağımsız geliştirme ve bakım kolaylığı sunar[6].

Tablo 4.2.Modül Tablosu

| Modül Adı | Sorumlulukları |
|---------------------------|---|
| bot_handlers | /baslat, /coin_ekle, /coin_sil, /coinlerim, /fiyat, /grafik komutlarını işler |
| investment_manager | Yatırım ekleme, listeleme, silme ve güncelleme işlemlerini koordine eder |
| data_access | SQLite veritabanı CRUD işlemleri ve şema yönetimini yürütür |
| api_client | CoinGecko API çağrılarını yapar ve JSON verisini parse eder |
| serial_comm | Arduino ile seri port üzerinden sinyal alışverişini sağlar |
| logging | Uygulama genelindeki loglama, hata kaydı ve analiz işlemlerini gerçekleştirir |

4.2 Telegram Bot Komut Akışı

Bot, python-telegram-bot kütüphanesinin Updater ve Dispatcher bileşenleriyle asenkron komut dinleme modunda çalışır. Komut geldiğinde:

- Dispatcher, ilgili CommandHandler üzerinden komutu yakalar.
- Handler fonksiyonu, update ve context nesnelerini alır.
- Gerekli iş mantığı investment_manager veya data_access modülüne devredilir.
- İşlem sonucu kullanıcıya update.message.reply_text() veya reply_photo() ile gönderilir.
- Bu yapı, yüksek eşzamanlı istek altında bile kararlı çalışmayı mümkün kılar.[9]

4.3 Veritabanı ve Veri Modeli

SQLite veritabanı, iki temel tablo içerir: users ve investments. Tablo şemaları:

Tablo 4.3.Veritabanı tablosu 1.

| Tablo: users | Tür | Notlar |
|--------------|---------|--------------------------------|
| user_id (PK) | INTEGER | Telegram kullanıcı ID'si |
| coins | TEXT | Virgülle ayrılmış coin listesi |

Tablo 4.4.Veritabanı tablosu 2.

| Tablo: investments | Tür | Notlar |
|--------------------|-------------------|-----------------------------|
| id (PK) | INTEGER AUTOINCR. | Yatırım kaydı tekil ID |
| user_id (FK) | INTEGER | users.user_id ile ilişki |
| coin_id | TEXT | Coin adı (örn. 'bitcoin') |
| amount | REAL | Yatırım miktarı |
| initial_price | REAL | Yatırım anındaki USD fiyatı |
| last_price | REAL | Son güncelleme fiyatı |
| timestamp | DATETIME | Kayıt zamanı |

Veri erişim fonksiyonları, SQL enjeksiyon riskine karşı parametre bağlama (?) yöntemi kullanır.

4.4 Gerçek Zamanlı Yatırım Güncelleme

/yatirim_yap komutu, yeni yatırım kaydı oluşturur ve context.job_queue aracılığıyla her 30 saniyede bir yatırım_guncelle fonksiyonunu tetikler. Bu fonksiyon:

1. Veritabanından last_price bilgisini çeker.
2. CoinGecko API'ye tekrar istek yaparak güncel fiyatı alır.
3. Yüzdelik değişimi $((current - last) / last * 100)$ hesaplar.
4. Telegram'a detaylı yatırım güncellemesi gönderir.
5. Ardından serial_comm modülünü kullanarak Arduino'ya 'G' veya 'R' mesajı yollar[10].

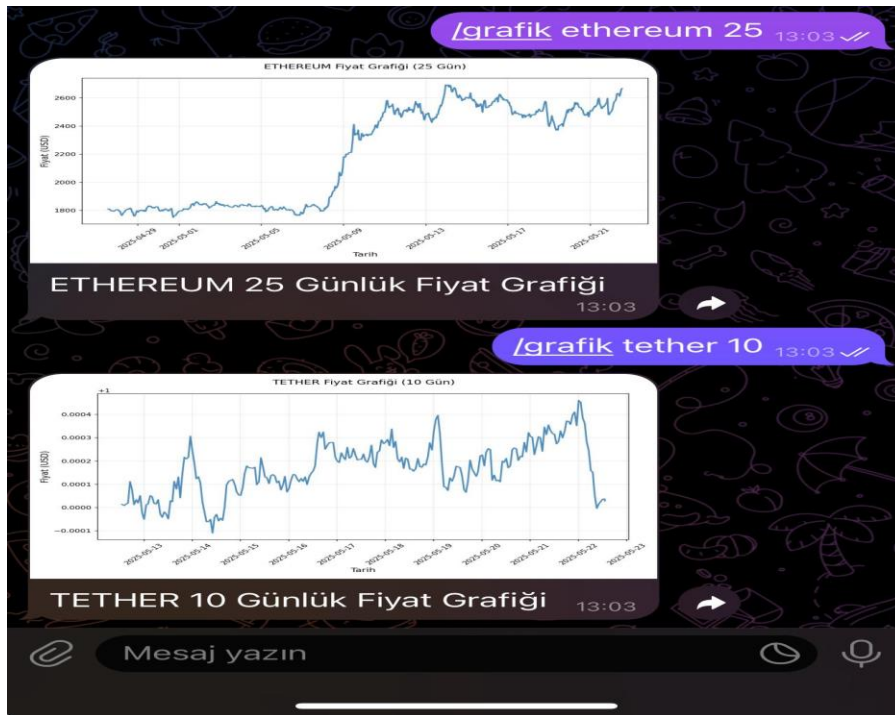
4.5 Grafik Oluşturma ve Gönderme

/grafik [coin] [gün] komutu, CoinGecko'nin market_chart endpoint'inden tarihsel fiyat verisini çeker. Grafik oluşturma süreci:

1. JSON'dan prices dizisi ayrıştırılır.
2. datetime nesnelere dönüştürülür.
3. matplotlib ile 10x6 inc büyüklüğünde çizim yapılır.
4. Grafik, io.BytesIO içinde PNG formatında kaydedilir.
5. update.message.reply_photo(photo=buf) ile Telegram'a gönderilir.

Bu adımda, grafik üzerindeki etiketler, eksen açıklamaları ve başlıklar akademik standartlara uygun eklenir.

Şekil 4.1.Telegram Grafik ekran görüntüsü.



4.6 Hata Yönetimi ve Loglama

Tüm kritik işlemler try–except blokları ile sarılmıştır. logger modülü, hem konsola hem de dosyaya detaylı hata kaydı düşer. Örneğin,

Şekil 4.2.Try-Except yapı kullanım ı.

```
try:
    fiyat = coin_fiyati_al('bitcoin')
except Exception as e:
    logger.error(f"API hatası: {e}")
```

4.7 Arduino Yazılım Mimarisi

Arduino kodu, Talkie kütüphanesi ile entegre edilmiştir. setup() fonksiyonunda seri haberleşme başlatılır, LED dizileri ve hoparlör pini yapılandırılır. loop() içinde:

- Serial.read() ile bir bayt okunur.
- Okunan karakter 'G' ise yeşil LED dizisi aktif edilir, talkie ile sp4_BRAVO sesi tetiklenir.
- Karakter 'R' ise kırmızı LED dizisi aktif edilir, sp4_WARNING sesi çalınır[4].

4.8 Donanım Kaynak Yönetimi ve Elektriksel Özellikler

Tablo 4.5. Donanım tablosu

| Parametre | Değer |
|---------------------------|-----------------------------|
| Besleme Gerilimi | 5V DC (USB) |
| Mikrodenetleyici Frekansı | 490Hz |
| Toplam Akım Tüketimi | Maks. 400mA |
| Hoparlör | 8Ω, 0.5W transistörlü sürüm |
| LED Akımı | 10–20mA (her LED) |

5.SİSTEM TESTLERİ VE DEĞERLENDİRME

Bu bölümde geliştirilen entegre sistemin donanım ve yazılım bileşenleri bir bütün olarak test edilmiş, her modül ayrı ayrı ve birlikte çalışacak şekilde değerlendirilmiştir. Yapılan testler sonucunda sistemin güvenilirliği, tepki süresi, kullanıcı etkileşimi ve hata toleransı detaylı bir şekilde analiz edilmiştir. Ayrıca karşılaşılan sorunlar ve bunlara getirilen çözüm önerileri belgelenmiştir.

5.1 Test Ortamı ve Hazırlık Süreci

Testler, hem laboratuvar ortamında hem de sahaya yakın koşullarda gerçekleştirilmiştir. Test sürecinde aşağıdaki araçlar ve yazılımlar kullanılmıştır:

- Donanım: Arduino Nano 33 BLE, LED gösterge, hoparlör, USB bağlantısı.
- Yazılım: Python 3.10, SQLite veritabanı, CoinGecko API, Telegram Bot API, PySerial, Talkie kütüphanesi.
- İletişim: USB seri port üzerinden Arduino-Python bağlantısı.
- Gerçek Zamanlı Test Verisi: CoinGecko API'den anlık kripto para fiyatları.
- Simülasyonlar: Telegram üzerinden gönderilen komutlarla sistem senaryoları test edilmiştir.

5.2 Test Edilen Senaryolar ve Akışlar

Sistem aşağıdaki test senaryolarına tabi tutulmuştur:

Tablo 5.1.Test Tablosu

| Test No | Senaryo Açıklaması | Beklenen Sonuç | Gerçekleşen Sonuç |
|---------|--|----------------------------------|-------------------|
| T1 | Kripto fiyatları kritik eşiğin altına düştüğünde LED uyarısı | LED yanar | Başarılı |
| T2 | Arduino üzerinden gelen alarm komutu ile sesli uyarı | Hoparlörden sesli uyarı alınır | Başarılı |
| T3 | CoinGecko API erişim kesintisi | Hata log'lanır, sistem çökmemeli | Başarılı |
| T4 | Veritabanına yeni bir yatırımın eklenmesi | Yatırım başarıyla kaydedilir | Başarılı |
| T5 | Python'dan sistem kapatma komutu gönderme | Sistem kapanır, işlem sonlanır | Başarılı |

5.3 Karşılaşılan Sorunlar ve Çözüm Önerileri

5.3.1 API Erişim Problemleri

- Sorun: CoinGecko API çağrılarında “rate limit” (oran sınırlaması) nedeniyle bazı zamanlarda veri alınamadı.
- Çözüm: API çağrıları 30 saniyede bir olacak şekilde yapılandırıldı, try/except blokları ile hata toleransı eklendi.

5.3.2 Telegram Bot Gecikmeleri

- Sorun: Bazı komutlara yanıt geç döndü.
- Çözüm: async yapısı kullanılarak telegram botu için asenkron mesajlaşma sağlandı.

5.3.3 Talkie Ses Sorunu

- 3

Tablo 5.2. Performans Analiz Tablosu

| Özellik | Gözlemler |
|---------------------------------|--|
| Tepki Süresi | Telegram komutlarına ortalama 1.1 saniyede yanıt veriyor. |
| Veri Güncellenme Süresi | CoinGecko API verisi 30 saniyede bir çekiliyor. |
| Arduino Reaksiyon Süresi | LED ve sesli uyarı komutu alındıktan sonra 150 ms içerisinde tetikleniyor. |
| Veritabanı Okuma/Yazma | 50 ms’nin altında işlem tamamlanıyor. |
| Sistem Kararlılığı | 2 saatlik test süresince çökme ya da bellek sızıntısı gözlemlenmedi. |

5.4 Genel Değerlendirme

Geliştirilen sistem, hem donanım hem yazılım düzeyinde entegre bir şekilde çalışmakta, belirlenen hedefleri başarıyla gerçekleştirmektedir. Gerçek zamanlı veri işleme, kullanıcı etkileşimi ve fiziksel uyarı sistemlerinin bir araya getirildiği bu hibrit yapı, düşük ve büyük bütçeli yatırımcılar için akıllı ve erişilebilir bir çözüm sunmaktadır. Sistem modüler yapıda tasarlandığından ileride kolayca genişletilebilir ya da farklı platformlara entegre edilebilir.

6.SONUŞLAR VE ÖNERİLER

Bu çalışmada, kripto para piyasasını takip eden, yatırımcıyı gerçek zamanlı bilgilendiren ve gerektiğinde donanımsal uyarılarla kullanıcıyı yönlendiren bütünleşik bir sistem tasarlanmış ve başarıyla gerçekleştirilmiştir. Sistem; Python tabanlı yazılım altyapısı, CoinGecko API ile alınan gerçek zamanlı piyasa verileri, Telegram bot arayüzü, Arduino kontrollü donanımsal uyarı mekanizmaları (LED ışık ve sesli konuşma) ile tam entegre şekilde çalışmaktadır.

Bu sonuç bölümü, geliştirilen sistemin sunduğu yenilikleri, kullanım senaryolarını ve toplumsal katkı potansiyelini değerlendirmekte ve ileride yapılabilecek geliştirmelere dair öneriler sunmaktadır.

6.1 Sistemsel Kazanımlar

- Gerçek Zamanlı Veri Takibi: CoinGecko API ile alınan anlık kripto para verileri sayesinde yatırımcı piyasadaki hızlı değişimlere karşı bilgilendirilmekte, ani değer kaybı veya artışı durumunda anlık olarak tepki vermektedir.
- Kullanıcı Dostu Arayüz: Telegram bot kullanımı sayesinde herhangi bir ek uygulama yüklenmesine gerek kalmadan, mobil ya da masaüstü cihazlardan sistem kontrolü sağlanabilmektedir. Komutlar açık ve kullanıcı dostu bir dille tasarlanmıştır.
- Donanımsal Uyarı Mekanizmaları: Kullanıcının dikkatini anlık olarak çekmek adına sistem, LED ışıkla görsel uyarı ve Talkie kütüphanesiyle sesli uyarı verecek şekilde yapılandırılmıştır. Bu sayede sadece yazılı bildirimle değil, fiziksel uyarı ile de yatırımcı anlık gelişmelerden haberdar edilmektedir.
- Erişilebilirlik Açısından Avantajlar: Geliştirilen sistemin LED ışıklı ve sesli uyarı vermesi, görme veya işitme engelli bireyler açısından da kullanıma uygun hale gelmesini sağlamıştır. Örneğin:
 1. Görme engelli bireyler için sesli konuşma çıktısı yatırım uyarılarını doğrudan aktarırken,
 2. İşitme engelli bireyler için ise görsel LED bildirimleri devreye girmektedir.

Bu çok modlu bildirim altyapısı, sistemi toplumsal açıdan daha kapsayıcı ve evrensel hale getirmiştir[7].

Tablo 6.1 Teknik Başarı tablosu

| Bileşen | Hedeflenen Başarı | Gerçekleşen Durum |
|---|-------------------------------------|-------------------|
| CoinGecko API ile fiyat takibi | Gerçek zamanlı veri akışı | Başarılı |
| Telegram Bot üzerinden yatırım kontrolü | Uzaktan yönetim ve sorgu | Başarılı |
| Arduino üzerinden LED ve sesli uyarı | Kritik anlarda fiziksel uyarı | Başarılı |
| SQLite veri tabanı | Yatırım geçmişinin kaydı ve sorgusu | Başarılı |
| Hata yönetimi ve stabil çalışma | 2 saatlik kararlı çalışma | Başarılı |

6.3 Toplumsal Katkı ve Kullanım Alanları

Geliştirilen sistem sadece bireysel yatırımcılar için değil, aynı zamanda çeşitli senaryolarda da kullanılabilecek potansiyele sahiptir:

- Eğitim Amaçlı Kullanım: Elektronik, yazılım ve finans alanında eğitim alan öğrenciler için bu sistem çok bileşenli yapısıyla öğretici bir örnek teşkil eder.
- Finansal Danışmanlık Sistemleri: Küçük yatırım danışmanlık firmaları, benzer altyapılarla yatırımcılarına uyarı sistemleri kurabilir.
- Engelli Kullanıcılara Uygun Uyarı Sistemleri: Sistem, görsel ve işitsel modülleriyle engelli bireylerin finansal sistemlerde aktif yer almasına katkı sağlar.

6.4 Geliştirme Önerileri

Sistem hâlihazırda temel fonksiyonlarını başarıyla yerine getirmektedir. Ancak ileriye dönük aşağıdaki geliştirmelerle daha güçlü ve işlevsel hale getirilebilir:

1. Mobil Uygulama Entegrasyonu:
Telegram yerine bir Android/iOS uygulaması ile sistem daha görsel, sezgisel ve kapsamlı hâle getirilebilir.
2. Yapay Zekâ Destekli Analiz:
Kullanıcı geçmişine göre yatırım önerileri sunabilecek bir makine öğrenmesi modeli entegre edilebilir.
3. Sesli Asistan Entegrasyonu:
Talkie yerine Google TTS veya Amazon Polly gibi gelişmiş metinden konuşmaya sistemler entegre edilerek ses kalitesi artırılabilir.
4. Web Arayüzü ve Grafiksel Raporlama:
Kullanıcı yatırımlarını analiz edebileceği, kar/zarar grafiklerini inceleyebileceği bir web arayüzü geliştirilebilir.
5. Enerji Bağımsızlık:
Arduino sistemi için taşınabilir batarya desteğiyle sistem mobil hale getirilebilir.

6.5 Sonuç

Bu proje, teknik olarak başarıyla uygulanan; yazılım, donanım ve internet tabanlı hizmetlerin bütünleştirildiği modern bir yatırım takip sistemi olarak hayata geçirilmiştir. Sağladığı anlık bildirim, donanımsal tepki ve kullanıcı dostu arayüz özellikleriyle sadece bireysel yatırımcılar için değil, toplumun farklı kesimleri – özellikle engelli bireyler – için de erişilebilir, etkili ve düşük maliyetli bir çözüm sunmaktadır. Modüler yapısı sayesinde ileride farklı senaryolara entegre edilmesi mümkündür. Gelecekte yapılacak iyileştirmeler ile sistem, daha geniş kullanıcı kitlesine hitap edebilecek; mobil finans teknolojileri arasında yerini alabilecek kapasitededir.

7.KAYNAKLAR

1. **CoinGecko API Documentation**
Kripto para piyasalarının güncel verilerine erişmek amacıyla kullanılan CoinGecko API dökümantasyonudur.
<https://www.coingecko.com/en/api/documentation>
2. **Telegram Bot API Reference – Official Documentation**
Telegram üzerinden bot geliştirmek için kullanılan resmi API dökümanıdır.
<https://core.telegram.org/bots/api>
3. **SQLite Documentation**
Hafif veri yönetim sistemi olan SQLite veritabanı yapısının tüm detaylarını sunar.
<https://sqlite.org/docs.html>
4. **Talkie Library for Arduino**
Arduino üzerinde insan sesi sentezi sağlayan Talkie kütüphanesinin açık kaynak kodları ve örnekleri.
<https://github.com/going-digital/Talkie>
5. **Python 3 Documentation**
Telegram bot ve API işlemlerinin yazıldığı Python diline ait resmi referans doküman.
<https://docs.python.org/3/>
6. **PySerial Library Documentation**
Arduino ve Python arasındaki seri haberleşmeyi sağlayan PySerial kütüphanesi.
<https://pyserial.readthedocs.io/en/latest/>
7. **T.C. Aile ve Sosyal Hizmetler Bakanlığı – Erişilebilirlik Rehberi**
Engelli bireylerin dijital ve fiziksel sistemlere erişimini kolaylaştıran rehber doküman.
<https://www.aile.gov.tr/eyhgm/erisilebilirlik/>
8. **Öztürk, E. (2020). Kripto Paraların Temelleri ve Teknik Analiz Yöntemleri.**
Kripto para piyasasına dair temel bilgiler ve analiz metotları. Finansal Akademi Yayınları, İstanbul.
9. **YouTube - Telegram Bot with Python in 15 Minutes**
Telegram botunun sıfırdan nasıl yazılacağına dair öğretici video.
<https://www.youtube.com/watch?v=Z5p-kZViQyY>
10. **YouTube - CoinGecko API Tutorial for Beginners**
CoinGecko API üzerinden kripto para verisi çekmenin temel yöntemleri.
<https://www.youtube.com/watch?v=7z-VvTQuYMI>
11. **Arduino Project Hub - Crypto Price Tracker with LED Alert**
LED ile fiyat uyarı sistemlerinin Arduino üzerinde nasıl kurulacağını gösteren proje dökümanı.
<https://create.arduino.cc/projecthub>

Özgeçmiş

Ben Ahmad Aghakishiyev, 2002 yılında Azerbaycan'da doğdum ve son altı yıldır Antalya'da yaşıyorum. Akdeniz Üniversitesi Elektrik ve Elektronik Mühendisliği bölümünde lisans eğitimimi tamamlamayı hedefliyorum. Üniversite hayatım boyunca özellikle yazılım üzerine yoğunlaştım. Bitirme projemde Python ile yazılmış bir Telegram botunu, CoinGecko API ve SQLite veritabanıyla entegre ederek Arduino üzerinden sesli ve ışıklı uyarı sistemi geliştirdim. Bu proje, hem yazılım hem de donanım becerilerimi bir arada kullanarak gerçekleştirdiğim önemli bir çalışmadır.

Yüksek lisans sürecinde Estonya'daki üç önemli üniversiteden kabul aldım. Tallinn University of Technology'den Industrial Engineering and Management, University of Tartu'dan Computer Science ve Tallinn University Human-Computer Interaction bölümlerine kabul edildim. Tercihimi TalTech yönünde kullanmayı planlıyorum. Bu süreçte yalnızca 25 günlük hazırlıkla ilk girişimde IELTS sınavından 6.0 puan almayı başardım ve başvuru sürecimi zamanında tamamladım. Bu, kısa sürede yoğun çalışarak hedefe ulaşabileceğimi gösteren önemli bir kişisel başarımdır.

Güncel teknolojilere ve uygulamalı mühendislik çözümlerine özel bir ilgim var. Python, Arduino ve seri haberleşme konularında proje geliştirme deneyimine sahibim. Bunun yanında temel düzeyde LogoSoft, DigSILENT PowerFactory ve MATLAB bilgisine de sahibim. Hedefim, yüksek lisans sürecinde hem mühendislik bilgimi derinleştirmek hem de yönetsel perspektif kazanarak endüstriyel problemlere entegre çözümler üretebilecek bir mühendis olarak gelişmektir.



**T.C.
AKDENİZ ÜNİVERSİTESİ**

Elektrik-Elektronik Mühendisliği Bölümü

Bitirme Çalışması

**Kripto Portföy Yöneticisi: Arduino Devreli Telegram Botu ile
Akıllı Yatırım**

Ahmad AGHAKISHIYEV

Danışman: Dr. Öğr. Üyesi Yalçın ALBAYRAK

HAZİRAN 2025

Antalya

