

# Zeiterfassung-App



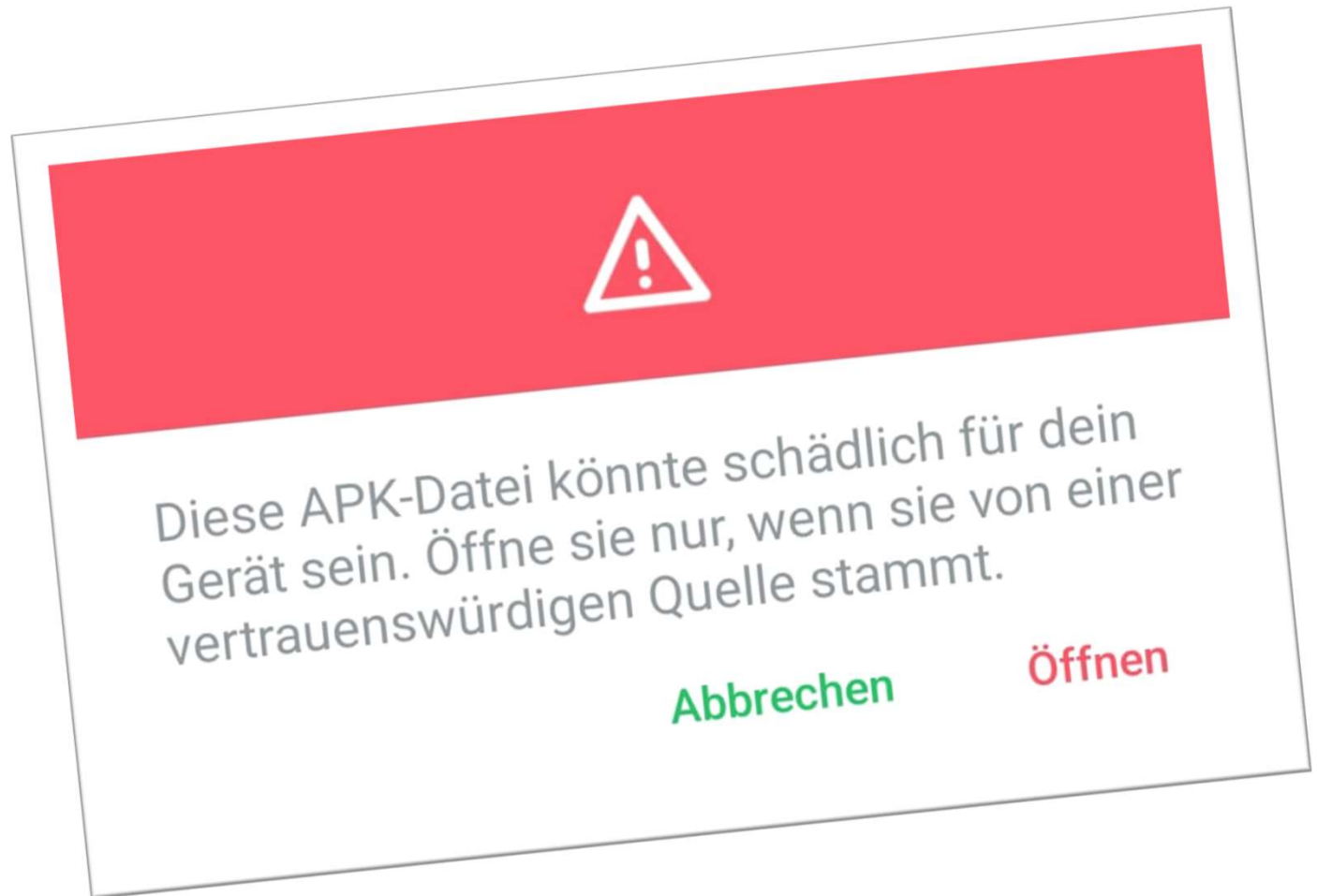
Entwickelt von: *Ashraf Yahya*

Betreuer: *Prof. Dr. Markus Kampmann*

# Warnung!

---

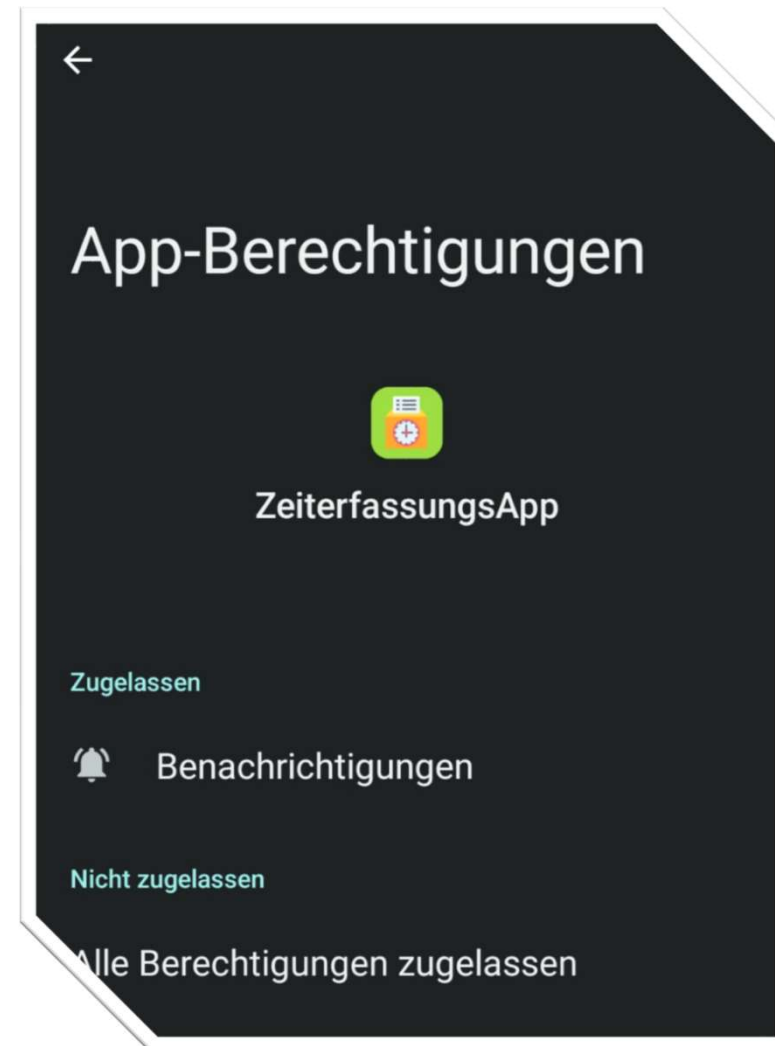
- Debug-Version
- Durch WhatsApp



# Berechtigung!

---

- Berechtigung zulassen



# Hintergrund und Ziel

- Die Zeiterfassungs-App wurde entwickelt, um eine einfache und effiziente Lösung zur Erfassung und Verwaltung von Arbeitszeiten zu bieten.
- Sie richtet sich sowohl an Nutzer, die ihre Arbeitszeiten erfassen wollen, als auch an Administratoren, die diese Daten überwachen und verwalten.

# Bedeutung der Zeiterfassung

---

- **Warum ist Zeiterfassung wichtig?**
- Zeiterfassung ist entscheidend für die Überwachung der Arbeitszeiten, das Management von Projekten und die Einhaltung gesetzlicher Vorgaben.
- Unternehmen können so Überstunden berechnen, die Produktivität steigern und gesetzliche Anforderungen einhalten.

# Verwendete Technologien

---

- **Android Studio:** Hauptentwicklungsumgebung zur Erstellung der App.
- **Java:** Programmiersprache für die Entwicklung der App-Logik.
- **Firebase:** Backend-Dienst für Authentifizierung, Datenhaltung und Echtzeit-Synchronisation.

# Entwicklungsumgebung

---

## **Android Studio:**

- Installation und Einrichtung sind über die offizielle Android Studio-Website möglich.
- Wichtige Funktionen:
  - Code-Editor mit Syntax-Hervorhebung und Refactoring.
  - Layout-Editor für die Gestaltung der UI.
  - Debugging-Tools und Emulatoren.

# Architektur und Design

---

## Anwendungsarchitektur:

- **Client-Server-Modell:**
  - Client: Die mobile App auf Android.
  - Server: Firebase als Backend für Authentifizierung, Datenverwaltung und Verarbeitung.
- Datenfluss: Nutzer interagieren über die App mit Firebase, um Zeiteinträge zu speichern und abzurufen.



# Benutzeroberfläche

---

## UI-Design und Layout:

- **Klarheit und Einfachheit:** Intuitive und leicht zu bedienende Benutzeroberfläche.
- **Responsive Design:** Optimierung für verschiedene Bildschirmgrößen (Hochformat unterstützt, Querformat nicht vollständig).
- **Screenshots:** Hauptbildschirm, Login, Admin-Oberfläche.

# Backend-Design

## Firestore als Backend:

- **Datenstruktur in Firestore:**
  - Benutzerinformationen in der „users“-Sammlung.
  - Zeiteinträge in der „timeEntries“-Sammlung.
- **Sicherheitsregeln:** Zugriffskontrollen basierend auf Rollen und Authentifizierung.

# Wichtige Funktionen

---

## **Benutzeranmeldung und Zeiterfassung:**

- **Benutzeranmeldung:**

- Nutzer registrieren sich mit E-Mail und Passwort, die in Firebase gespeichert werden.

- **Zeiterfassung:**

- Check-In und Check-Out werden protokolliert.
- Echtzeit-Updates in der Datenbank.

# Admin-Funktionen

---

## Verwaltung und Rollen:

- **Zugriffskontrolle:** Administratoren können Benutzerrollen und Zeiteinträge verwalten.
- **Verwaltung von Benutzerdaten:** Administratoren können alle Benutzerdaten einsehen und bearbeiten.

# Implementierung

---

## Code-Beispiele und Layout:

- **MainActivity.java**: Hauptaktivität für Zeiterfassung und Anzeige von Zeiteinträgen.
- **AdminActivity.java**: Verwaltung der Benutzerrollen und Zeitübersicht.
- **Firebase-Konfiguration**: Authentifizierung und Datenhaltung in Firestore.

# Firestore-Konfiguration

---

## Integration von Firestore:

- **Firestore Authentication:** Ermöglicht sichere Anmeldung und Registrierung.
- **Firestore Cloud Firestore:** Speichert Benutzerdaten und Zeiteinträge.
- **Benachrichtigungen:** Zwei Arten von Benachrichtigungen: Push- und In-App-Messaging.

# Teststrategien

## Testmethoden:

- **Integrationstests:** Sicherstellung, dass alle Komponenten korrekt zusammenarbeiten.
- **Benutzerakzeptanztests:** Überprüfung der Benutzerfreundlichkeit durch Testnutzungen und Anpassungen basierend auf Feedback.

# Sicherheit und Datenschutz

## Sicherheitsüberlegungen:

- **Verschlüsselung:** Passwörter und sensible Daten werden verschlüsselt gespeichert.
- **Zugriffskontrollen:** Nur autorisierte Benutzer haben Zugriff auf bestimmte Funktionen und Daten.



# Wartung und Erweiterungen

---

## **Zukünftige Erweiterungen:**

- **Geplante Funktionen:**

- Berichterstellung für Nutzer und Admins.
- Überstundenberechnung und Anzeige.
- Speicherung von Anmeldedaten zur Erleichterung der Nutzung.
- Verbesserung der Responsivität und Unterstützung für Querformat.

# Fazit

## **Zusammenfassung:**

- Die Zeiterfassungs-App bietet eine benutzerfreundliche Lösung zur Erfassung und Verwaltung von Arbeitszeiten.
- Mit Android Studio, Java und Firebase wurde eine stabile und funktionale App erstellt.

# Schlussfolgerungen und Empfehlungen

---

## **Reflektion und Ausblick:**

- Der Entwicklungsprozess ermöglichte tiefere Einblicke in die Android-Entwicklung.
- Zukünftige Erweiterungen könnten die Benutzerfreundlichkeit und Funktionalität der App weiter verbessern.

# Danke!

Danke für Ihre Aufmerksamkeit!

Fragen?