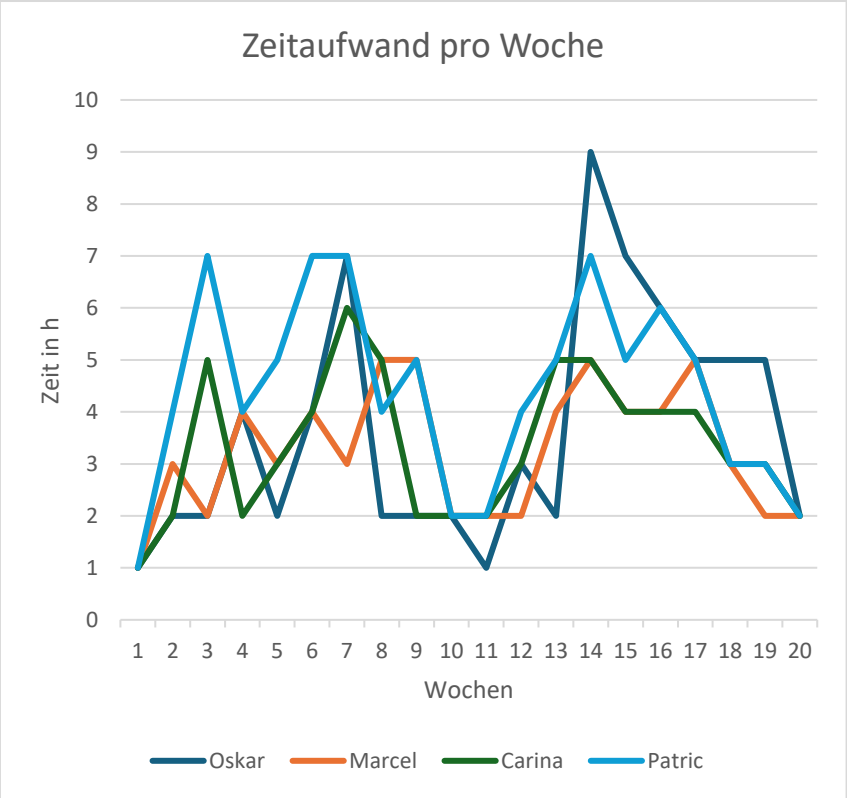
Gymtrackerx

Statistik:



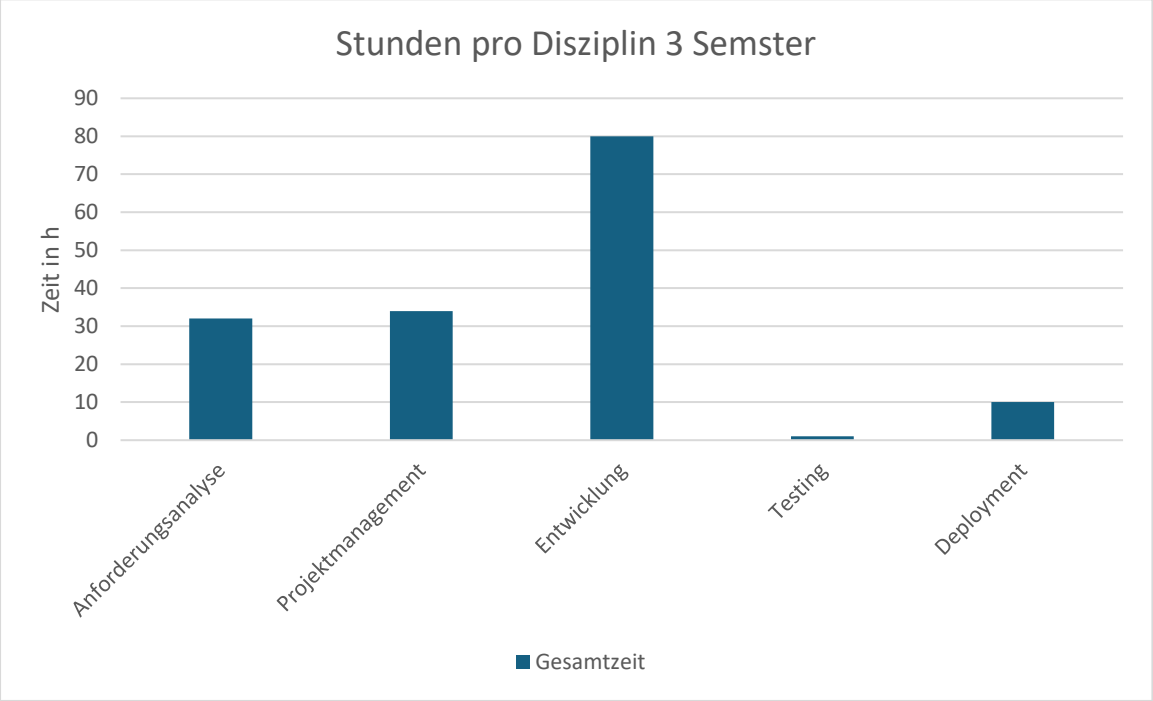
Hauptbeiträge:

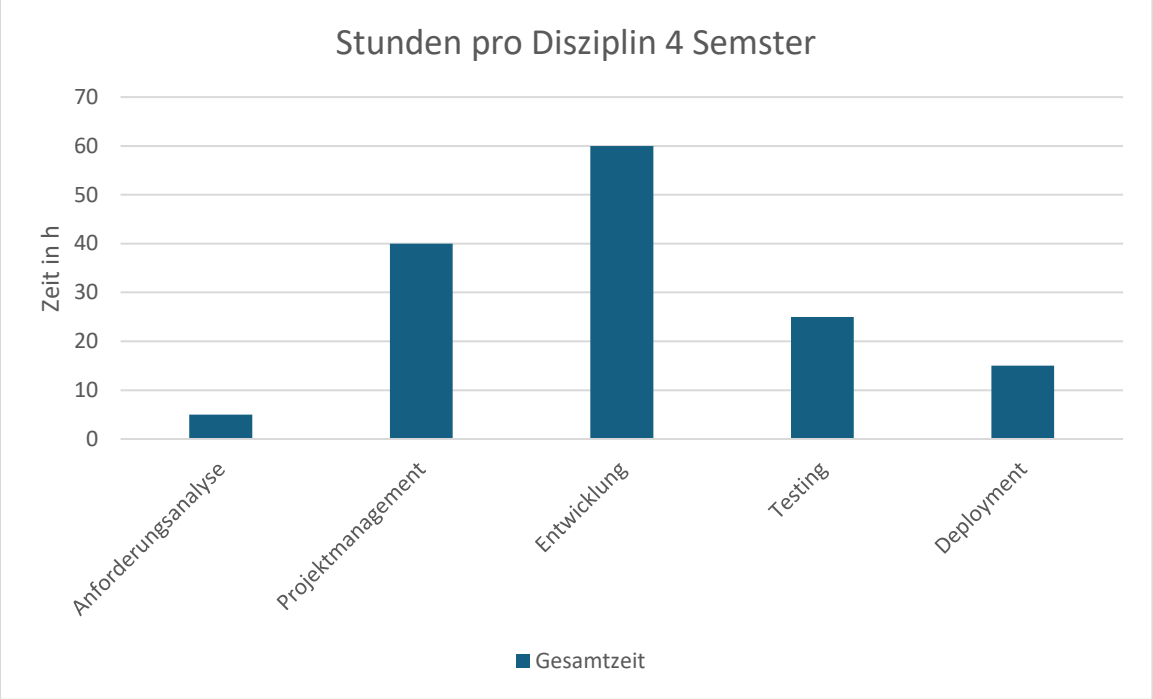
Oskar: Login, Register, Training, Exercise, Split & Workoutplandetail Screens und Funktionalität

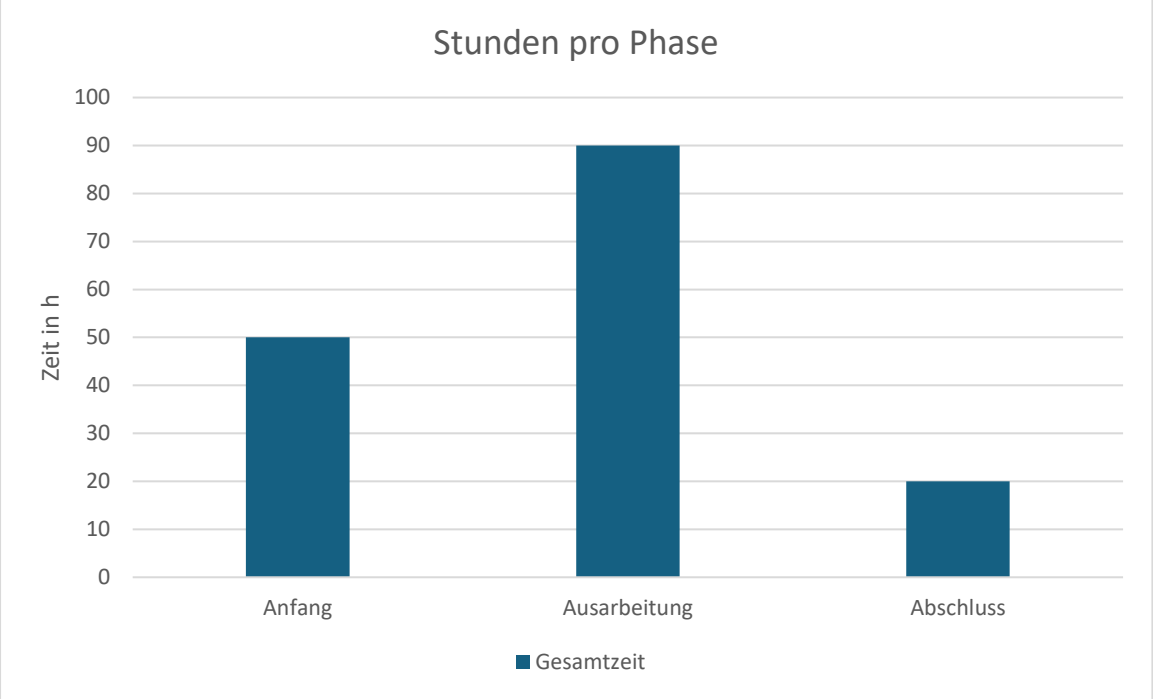
Marcel: UML, ADR, Seq.-Dia., Timer, arc24, Backlog

Carina: SRS, Planner Split Page, Qualitätsbaum, Jira-Zeitleiste und Disziplin+Phase, Metriken, Clean Code

Patric: Mockup, Main & Selection Screen, CI/CD, Login & Register backend, Datenbank, JWT




Stunden pro Disziplin 4 Semster
Zeit in h
70
60
50
40
30
20
10
0
Anforderungsanalyse
Projektmanagement
Entwicklung
Testing
Deployment
Gesamtzeit


Stunden pro Phase
Zeit in h
100
90
80
70
60
50
40
30
20
10
0
Anfang
Ausarbeitung
Abschluss
Gesamtzeit

CI/CD

Im Projekt wurden zwei CI/CD-Pipelines mit GitHub Actions eingerichtet: Eine für den main-Branch, die bei jedem Push oder Pull Request automatisch Linting, Format-Checks, Tests und einen Release-Build durchführt. Die andere für den dev-Branch erstellt bei Änderungen einen Debug-Build für Entwicklungszwecke. Beide Workflows laden das jeweilige APK als Artifact hoch. So wird Codequalität gesichert und der Build-Prozess automatisiert.

Architektur: Schichtenarchitektur

Die App verwendet eine Schichtenarchitektur, bestehend aus Frontend, Backend und Datenbank, um eine klare Trennung der Verantwortlichkeiten zu gewährleisten:

- Frontend: Realisiert mit Flutter für eine konsistente Benutzererfahrung auf verschiedenen Plattformen.
- Backend: Implementiert mit Node.js und Express, was flexible und skalierbare serverseitige Logik ermöglicht.
- Datenbank: MySQL bietet eine relationale Struktur, die ideal für die Speicherung und Verwaltung von Benutzerdaten, Trainingsplänen und Fortschrittsstatistiken ist.

Hauptargumente:

- Klare Trennung der Verantwortlichkeiten erleichtert Wartung und Erweiterung.
- Die strukturierte Datenverwaltung erlaubt einfache Datenanalyse und Optimierung der App-Funktionen.

Tech Stack

- Frontend: Flutter
- Backend: NodeJS, Express
- Frontend Backend Communication: RestAPI
- Datenbank: MySQL
- CI/CD: Github Actions
- Authentifizierung: JWT JSON Web Token
- Testing: Mocha, Flutter Tests
- IDE: Android Studio
- Projektmanagement Plattform: Jira
- Versions Verwaltung: Github
- Blog: Blog