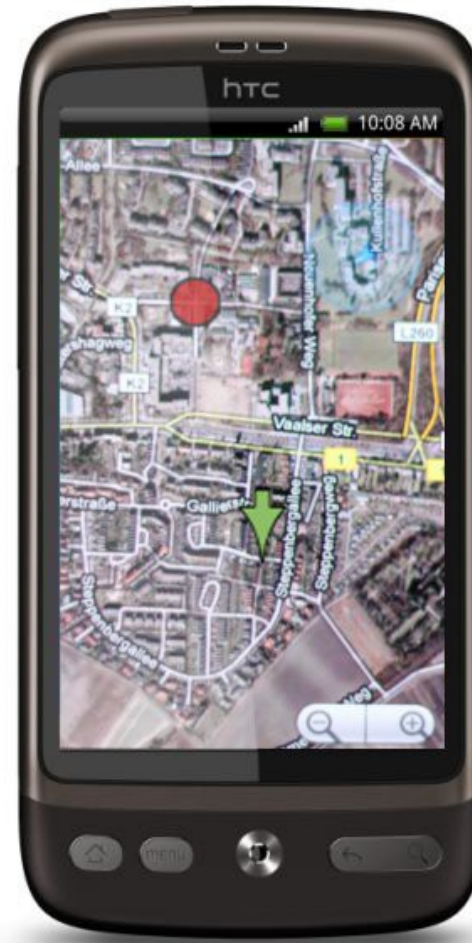


GeoCatch



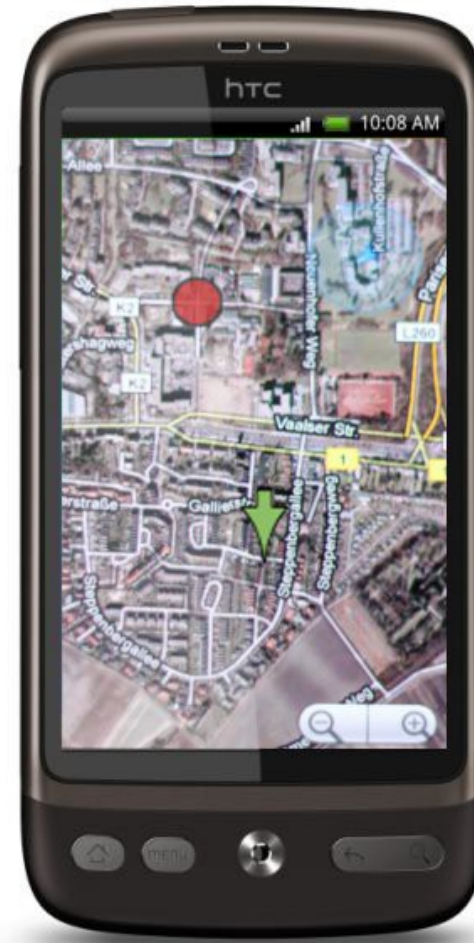
- Langeweile?
 - Jagdtrieb?
 - Mit Freunden treffen?
- => GeoCatch !



GeoCatch



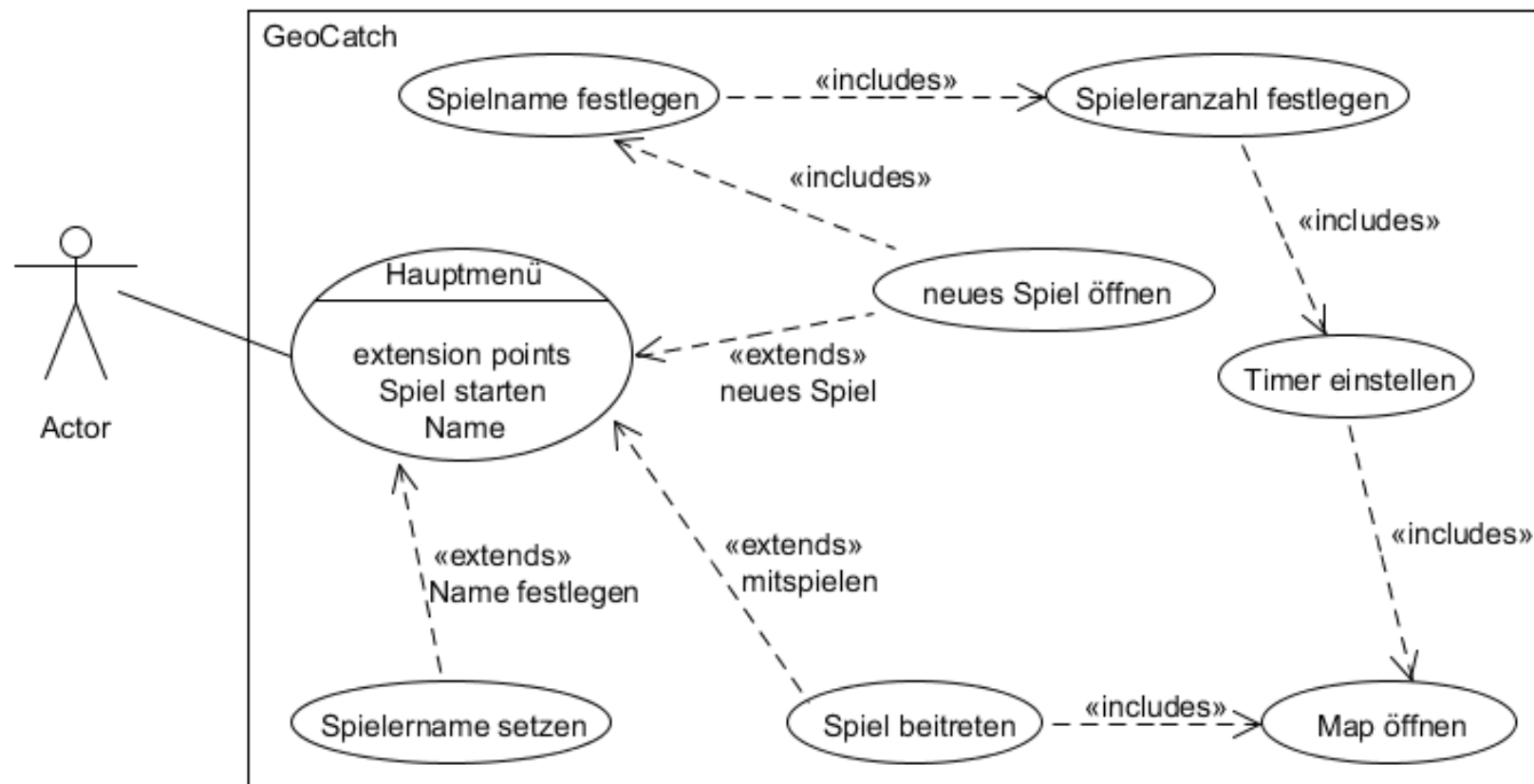
- Mind. 3 Spieler
- Sei Jäger und Gejagter
- Diverse Power-Ups
- Lobby-System



GeoCatch



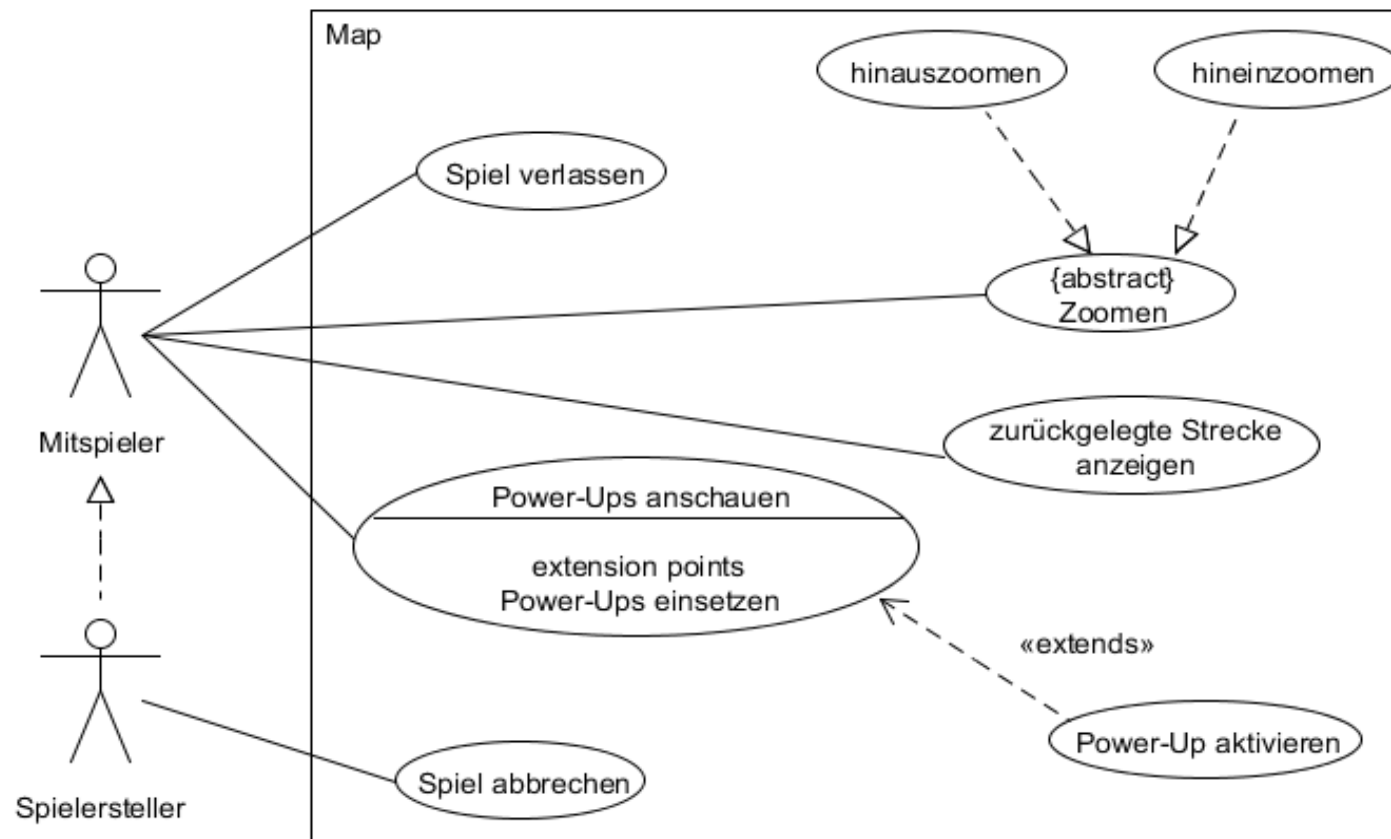
- Use Case Diagramm (Hauptmenu)



GeoCatch



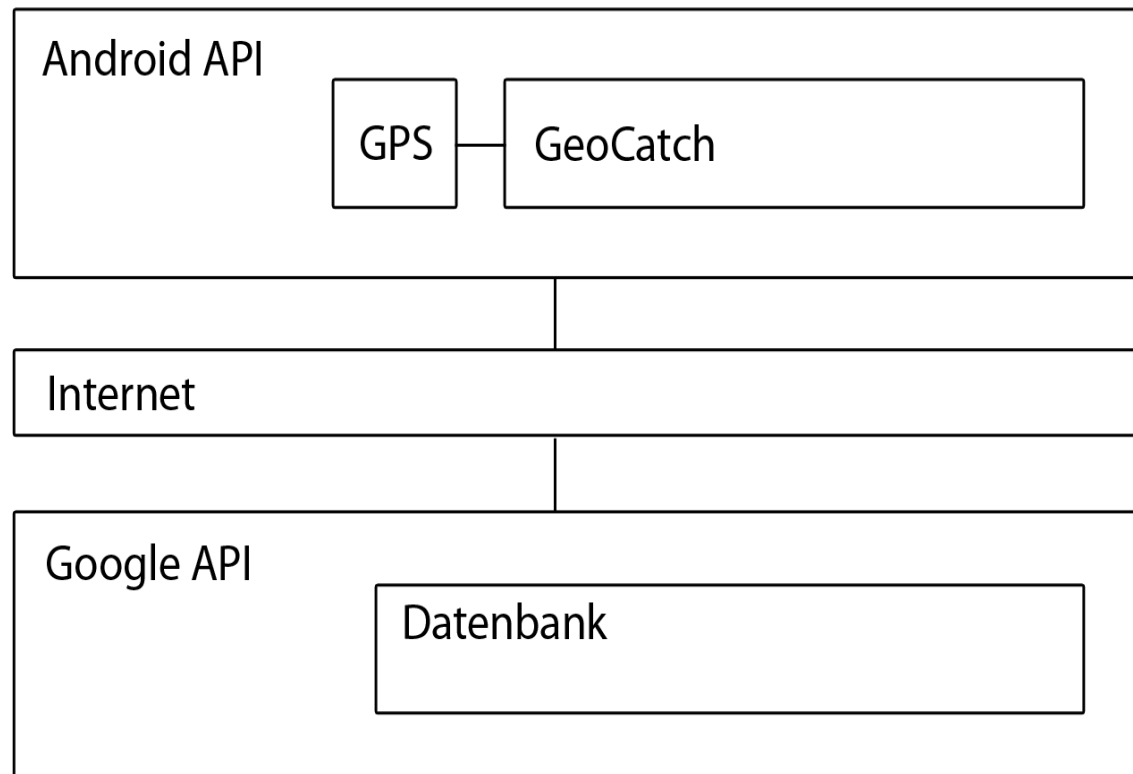
- Use Case Diagramm (Map)



GeoCatch



- Architektur



GeoCatch



Einzusetzende Technik

Frontend:

- Android 2.1 mit Google Maps API
- Java

Backend:

- Google AppEngine
- Python

GeoCatch



Evaluierungsplan

1. Debugging mittels Emulatoren
2. Mehrfache interne Testläufe auf echten Endgeräten
3. Anbieten im Android-Market
4. Feedback über Kommentare und Foren

GeoCatch



Zeitplan

Woche 1	Klassenentwurf, Schnittstellen festlegen, Einrichtung des Projekts(AppEngine, SVN), Lobby-System inklusive Start-Timer
Woche 2	Implementierung der eigentlichen Spiel-Logik => Anzeigen der eigenen Position und der Ziel-Position, Warnung bei Jäger in der Nähe, Spielende bei Fang
Woche 3	Erweiterung um Power-Ups (Anzeigen auf der Map, Menü zum Einsetzen der Power-Ups, Auswirkungen auf das Spiel)
Woche 4	Debugging, Testläufe, Veröffentlichung im Market und abschließende Evaluierung, sowie Kontakt mit Usern

GeoCatch



Aufgabenaufteilung

Christian	Analytiker => Anforderungen darstellen Software-Architekt => Spezifikationen festlegen
Sebastian	Integrierer => Zusammenspiel der Komponenten sichern Organisator => Verteilung der Aufgaben/Klassen
Stephan	Evaluierungs-Beauftragter => User Kontakt, Auswertung
Vladislav	Backend-Manager => Entwicklung/Wartung des Backends Grafiker