

# Don't break the chain

*Echipa Farapile (Ojog Cristian, Enciu Cristian, 341 C3)*

## Descriere detaliata proiect

### 1. Ideea generala:

Don't break the chain este o aplicatie de productivitate. Scopul ei este acela de a motiva utilizatorul sa-si indeplineasca task-urile propuse prin "positive reinforcement" primit de la prieteni. Prin "positive reinforcement" se intelege atat un sistem de puncte (tokeni / achievements) pe care utilizatorii le pot imparti intre prieteni. Tokenii vor fi primiti doar la completarea task-ului. Odata implementat sistemul de puncte putem introduce si leaderboard pentru ca utilizatorii sa poata avea "bragging rights" legat de cine este cel mai productiv.

### 2. General use case:

- userul creeaza un task (ex: vreau sa desenez cate ceva zilnic pentru a-mi face level up la skill-uri)
- in fiecare zi trebuie sa se tina de acel task (poate posta o dovada sau ceva.. depinde de tipul task-ului), altfel rupe lantul (fie o ia de la 0, fie i se adauga zile la target.. debatable)
- prietenii lui (de pe fb, g+, twitter) pot vedea task-urile lui si ii pot oferi o "incurajare" / token, pe care le poate primi userul cand isi indeplineste goal-ul (nu si mai devreme). incurajarile pot fi puncte sau chiar bani (paypal.. why not).
- cand ajunge la goal, poate share-ui succesul, sau poate cere incurajari pe parcurs.

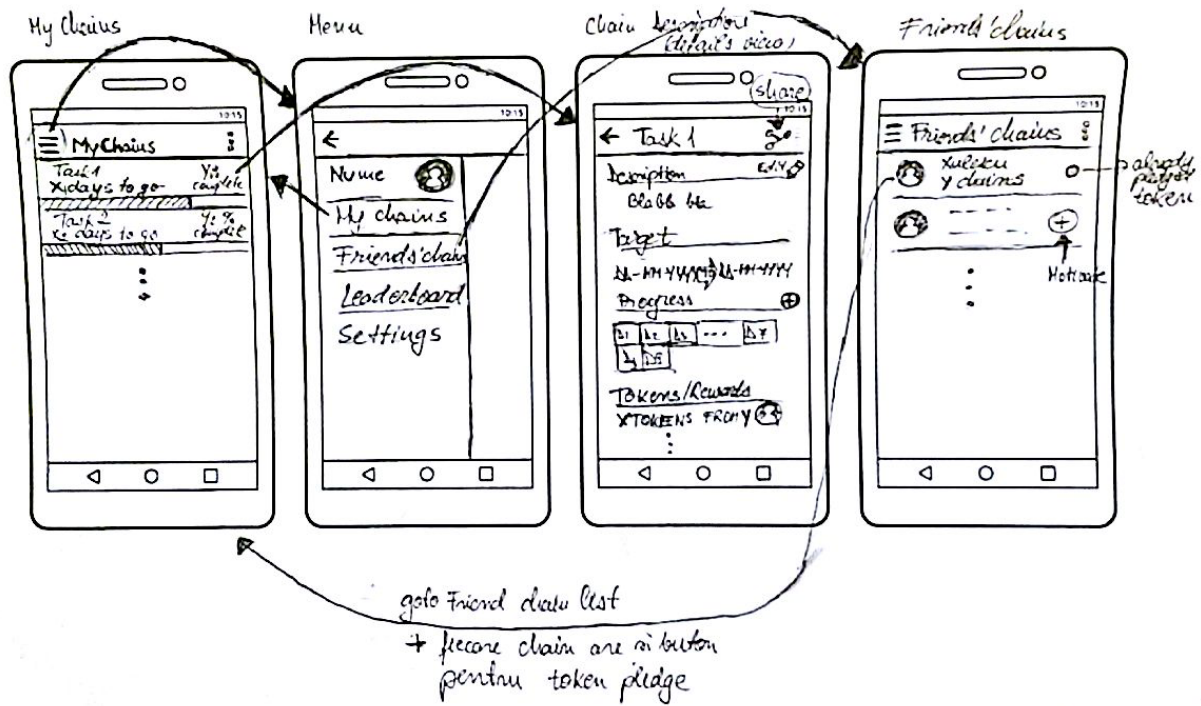
### 3. Tehnologii folosite:

Client - Android

IDE - Android Studio, pentru ca este mai bine suportat pe developers.android.com, si deja suficienta lume a facut pasul de la Eclipse incat sa existe resurse pe stackoverflow pentru orice problema

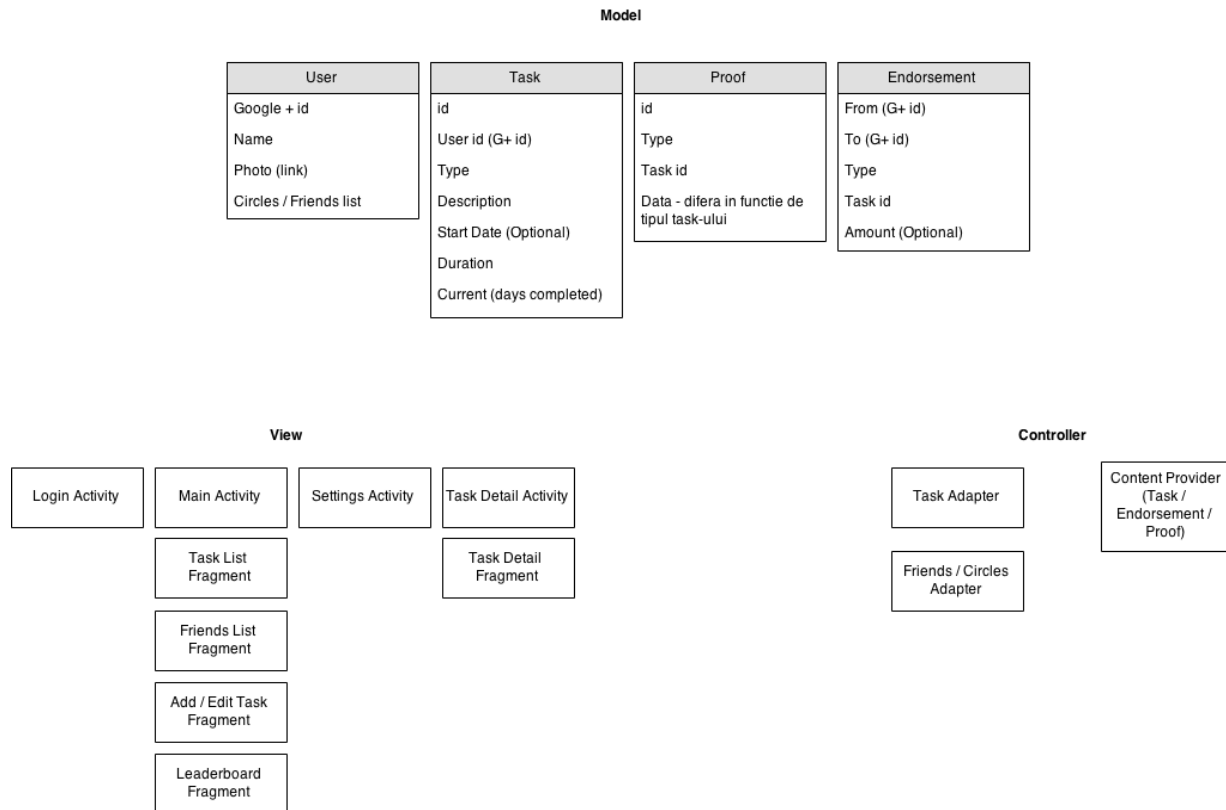
Backend - Google App Engine + Cloud Endpoints. Faciliteaza crearea unui API pe care il poate folosi clientul de Android. De asemenea, se poate adauga usor un modul de backend direct din Android Studio. (Google is nice like that)

# Wireframe



# Implementare

## Abordare Model - View - Controller



## Tehnologii:

Android Studio - app

Google App-Engine + Google Cloud Endpoints - backend: majoritatea backend-ului va constitui in query-uri si filtrari in baza de date, pentru a fi cat se poate de platform - independent. Practic app-ul ar trebui doar sa ia listele gata procesate prin content provideri. Cu toate acestea, app-ul trebuie sa mearga si fara ca utilizatorul sa faca login (poate nu vrea sa sharuiasca cu prieteni). De aceea, task-urile trebuiesc pastrate local si doar sincronizate cu backend-ul daca utilizatorul este logat.

NinjaMock - prototipare

## Milestones

## WP1 - Preliminary design

1. Definitivare wireframes
2. Stabilire arhitectura (class design pentru client si back-end)

## WP2 - State-of-the-art Technology Study

- ## 1. Studiu App-Engine + Cloud Endpoints pentru Android

### WP3 - Actual work

1. Functionalitate de baza back-end
2. Implementare front-end

## WP4 - Testing

1. Teste automate pentru backend
2. Teste manuale - intensiva pentru frontend

## WP5 - Publish

1. Publish App on Play store
2. Patching & Updates

## Timeline (Gantt)

[illegible]

