**Document de analiză a cerințelor clientului**

## Scopul aplicației:

Scopul principal al aplicatiei pe care o vom dezvolta este acela de a ajuta persoanele sa se simta in siguranta, indiferent in locul in care se afla.

## Obiectivele aplicației:

*Siguranta face parte din nevoile pe care o persoana le are, iar noi ne dorim sa realizam un sistem de securitate care sa permita clientului nostru sa se simta in control deplin atunci cand vine vorba de proprietatile pe care le detine. Deoarece asa cum am mentionat deja, controlul este un lucru important, pornim de la faptul ca pentru a te simti in control, in primul rand trebuie sa intelegi ce se intampla, astfel ca aplicatia noastra va fi una usor de utiliza.*

## Grupul țintă

*Aplicatia este adresata oricarei persoane care doreste sa se simta in siguranta.*

*Familiile isi vor putea monitoriza locuinta sau resedinta atunci cand nu se afla acasa, si nu numai. Ele vor putea avea acces la tot camerele de supraveghere si le vor putea activa/ dezactiva dupa bunul plac, avand si posibilitatea de a isi personaliza sistemul de securitate si de a il conecta cu lucrurile smart pe care le au in casa. Spre exemplu, pot opta pentru activarea camerelor de securitate doar atunci cand usa nu este incuiata din interior.*

*In ceea ce priveste cladirile de birouri, siguranta ar trebui sa reprezinte o nevoie de baza. Eu, ca utilizator al sistemelor de siguranta smart imi doresc sa pot vedea in timp real ce se intampla in birou, sa pot fi notificat in cazul in care cineva patrunde in cladirea de birouri intr-un anumit interval orar si de asemenea sa pot incuia de la distanta cladirea, in cazul in care cineva neatent a uitat sa faca asta.*

*Ca si persoana din sistemul sanitar, imi doresc sa pot avea certitudinea ca probele duse intr-un anumit laborator nu vor fi contaminate, astfel ca imi doresc sa monitorizez cu atentie ceea ce se intampla, chiar si in momentele in care nu sunt prezent fizic acolo.*

*Lista poate continua, intrucat grupul tinta al acestei aplicatii vizeaza orice persoana care isi doreste sa detina controlul asupra afacerilor si proprietatilor pe care le detine.*

## Colectarea cerințelor

*Care sunt cerințele pe care userii din story-urile de mai sus le-ar cere? Care sunt cerințele de sistem care apar ca o consecință a cerințelor userului (performanța în anumiți parametrii; utilizarea anumitor resurse externe; utilizarea unui mod specific de dezvoltare ș.a.m.d.). Această zonă este bine completată dacă are un număr cât mai mare de cerințe (relevante), chiar dacă sunt prea multe, sau depășesc un pic dimensiunea proiectului pe care îl avem în vedere. Aici trebuie să înțelegem tot ce* ***s-ar putea*** *face.*

## Interpretarea și prioritizarea cerințelor

*Dintre cerințele de mai sus vom interpreta și prioritiza cerințele.*

*1. Label-uiți cerințele funcționale / non-funcționale. Cerințele funcționale sunt cele care îndeplinesc o nevoie care a reieșit dintr-un user story, și răspund la întrebarea ce trebuie aplicația să facă. Cerințele nonfuncționale sunt cele care descriu calitățile de sistem, și răspund întrebărilor de tipul cum trebuie să fie un anumit feature sau aplicația cu totul? Acest label-ing e suficient să îl faceți în documentul de analiză, nu e necesar să îl cărați pe mai departe.*

***Functionale:*** *Armare / Dezarmare casa, Notificare in caz de alarmare, alarmare automata, dashboard all time.*

***Non functionale:*** *Api pentru comenzile de user, ceva chron pt alarmare automata, api notificari.*

*2. Gruparea cerințelor*

*Identificați criterii de gruparea cerințelor care ulterior credeți că vă vor ajuta la dezvoltare.*

*- Folosiți criterii pentru a grupa cerințele într-un mod care vi se pare util – după zona de tehnologie (BE, DevOps ș.a.) după eventualele module ale app (comunicare, procesare, stocarea datelor ș.a.), după feature-uri. Menționați aceste categorii și în documentul de analiză.*

***Dezvoltare (implementare features), devops (CI/CD, api notificari)***

*3. La acest moment puteți să creați un repo de GitHub pentru proiectul vostru și să puneți cerințele într-un back-log de issue-uri în GitHub Issues. Folosiți grupările de la 2 pentru a crea label-uri relevante pentru fiecare issue.*

*Resources:*[*The Product Backlog: A Step-by-step Guide*](https://www.toptal.com/product-managers/agile/product-backlog-step-by-step-guide) *În general sunt bune articolele de la toptal pe software engineering.*

*4. Play planning poker. Planning poker e un joc prin care toți membrii echipei evaluează prioritatea și dificultatea taskurilor în raport cu abilitățile individuale. Media acestor evaluări generează nivelul de prioritate și dificultate final al cerinței.*

*Recomand aplicația de aici: scrumpoker.online (pare că are și o integrare cu GitHub, dar nu am testat-o)*

*Pentru fiecare issue, veți juca câte două runde de planning poker. Trebuie să fiți într-un call. După ce alegeți issue-ul, fiecare user se autentifică în aplicație, și simultan ar trebui să votați dificultatea acelui issue. Faceți media și notați-o. Discutați dacă considerați că este cazul, și feel free să schimbați rezultatul dacă vi se pare că nu e potrivit.   
Repetați pentru a identifica prioritatea taskului (adică cât de valoros este pentru aplicația finală). Notați și acest rezultat la fiecare task.  
  
5. Plot the issues.*

*Realizați o axă, unde pe una dintre axe aveți dificultataea, iar pe cealaltă, prioritatea. Împărțiți axa în 4 cadrane (usor-valoros, dificil-valoros, usor-nevaloros, dificil-nevaloros). Astfel toate cerințele vor fi grupate în 4 categorii. Veți prioritiza cerințele usor-valoros, veți pune în backlog cele usor-nevaloroase și fidicil-valoroase, și în nice-to-have cele dificil nevaloros.*

*6. Check the features.*

*Verificați că adunate toate issue-urile prioritare, și din backlog, adunat constituie cele 5 feature-uri minim necesare pentru a îndeplini proiectul.   
  
7. Add technical issues.*

*Adăugați issue-uri doar în GitHub de care va trebui să vă ocupați care nu reies din cerințe (dev setup, deployment setup, etc.)*

## Alocarea rolurilor

*Fiecare membru va avea assigned pe trello un task dorit.*

## Documentul de analiză va fi adăugat în GitHub-ul proiectului.