Índex

# Introducció

## Context

## Formulació del problema

## Actors

# Estat de l’art

# Planificació inicial

# Planificació final

# Metodologia

# Sostenibilitat

# Conclusions

# Principis de Resiliència

Començar pels problemes que ens trobem en el software, errors propis, errors de dependències, (ja en àmbit del les aplicacions mòbils) errors pels terminals, errors per circumstancies causades per la mobilitat com ara: pèrdua de connexió a internet, o pèrdua de connexió gps, el fet que l’ús de l’aplicació podria ser interromput en qualsevol moment per una trucada, etc. per l’ús que els hi dona l’usuari. Aquesta és una pobre classificació dels possibles errors o circumstancies inesperades en que pot arribar Els principis estudiats, analitzats proposats i els futurs implementats -> introducció al següent capítol

# Pla del projecte

La planificació, estimació de costos, desviacions i mesures, avaluació riscos, costs i probabilitats , l’impacte socio-economico-ambiental

# Execució del projecte

La implementació: descripció de l’aplicació que hem construït, funcionalitats bàsiques, i les implicacions, els problemes (referencies creuades als problemes vists en l’apartat 2) que apliquen i les solucions implementades, nivells de resiliència, cas ideal, cas implementat, contractes, desacoblament i nivell d’intrusió(intrusisme).

Acabem amb conclusions pràctiques de la implementació dels principis. La composició dels tests, automatitzats. Analitzar resultats, comportament de l’aplicació amb la capa de resiliència i sense.

Explicació detallada dels casos d’us

# Conclusions

Conclusions de la planificació, teòrica dels principis que considerem com a proveïdors de resiliència en el software descripció de l’aplicació que hem construït, autocrítica,

Treball futur i possibles ampliacions.

Disseny en micro artefactes – micro funcionalitats, determina el disseny – arq de microserveis, en lloc de contenidors grans.

I.A – regressió de series números, polinomi de dos graus, a partir de 100 – Deeplearning4j, la xarxa pren el control, capa de resiliència – iOT > corregir sensors,

Meditacions sobre resiliència: mentre estic implementant

Estil escèptic - fred, independent - sense justificacions i amb seguretat!

Temes: nivells

La solució ideal depèn dels casos en els que ens trobem. Considerem dos grans casos. El software ja esta construït o el procés de la seva construcció no ha començat. En el segon cas es pot donar una millor i més amplia solució. Tant si s’aplica la resiliència a nivell intrusiu com si només s’implementa el contracte necessari per la posterior aplicació d’una capa externa i desacoblada de resiliència.

En el segon cas és pot aplicar un cert nivell de resiliència mitjançant solucions externes, però, sempre que l’aplicació compleixi les condicions requerides pels diferents elements: *dokeos* que aporten resiliència a un programari.

El present treball té l’ambició de suggerir principis aplicables en els casos esmentats o en qualsevol altre estat entremig. Només caldrà fer l’anàlisi dels contractes necessaris o dels canvis necessaris.

Encara que no pretenem fixar l’aplicació dels principis de resiliència, en metodologies àgils la resiliència podria generar un backlog propi o simplement afegir històries d’usuari per la creació dokeos.

Resiliència, es pot encarregar de gestionar i fer complir les pre condicions de les funcionalitats d’un software.

Cal tenir en compte que el conjunt de dokeos implementats aquí són caracteristics per a aplicacions mòbils. La gran majoria dels principis serveixen per qualsevol programari però els dokeos implementats no estan desacoblats d’Android.

Paraules clau:

dokeo - la implementació d’un element que prové d’un principi teòric de resiliència.

dokimos - conjunt de dokeos que proporcionen la resiliència d’una aplicació.