Pet Shop and Pet Rescue Center

Version <2.1>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 26/03/2023 | 1.0 | Creare document | Denisa-Maria Herlea |
| 02/04/2023 | 2.0 | Actualizare document | Denisa-Maria Herlea |
| 18/04/2023 | 2.1 | Actualizare document | Denisa-Maria Herlea |
|  |  |  |  |

Table of Contents

1. Introduction 4

2. Non-functional Requirements 4

2.1 Availability 4

2.2 Performance 4

2.3 Security 4

2.4 Testability 4

2.5 Usability 4

3. Design Constraints 4

# Introduction

Cerințele non-funcționale ale proiectului sunt la fel de importante ca și cerințele funcționale. Aceste cerințe se referă la calitățile sistemului și includ aspecte precum disponibilitatea, testabilitatea și ușurința de utilizare. Alte cerințe non-funcționale pot include performanța sistemului, securitatea, constrângerile de design.

# Non-functional Requirements

## Availability

Definiție: Capacitatea sistemului de a fi disponibil și funcțional pentru utilizatori în timpul activităților lor planificate.

Sursa stimulului: Utilizatorul.

Stimul: Solicitarea de accesare a sistemului.

Mediu: Condiția actuală a sistemului când solicitarea de acces ajunge la acesta.

Artefact: Intregul sistem sau părți ale acestuia care reacționează la solicitare.

Răspuns: Furnizarea accesului solicitat într-un interval de timp decent.

Măsură de răspuns: Timpul când sistemul este disponibil și funcțional pentru utilizatori.

Tactici: Redundanță, recuperare după eșec.

## Performance

Definiție: Viteza și timpul de răspuns al sistemului.

Sursa de stimul: Utilizatorul sistemului.

Stimul: Utilizatorul face clic pe un buton sau încearcă să acceseze o pagină.

Mediu: Accesarea sistemului in general.

Componentă: Sistemul web

Răspuns: Sistemul trebuie să furnizeze utilizatorului o pagină cu informațiile căutate în scurt timp.

Măsură de răspuns: Timpul de încărcare a paginii.

Tactici: Comprimarea fișierelor, optimizarea bazei de date.

## Security

Definiție: Capacitatea sistemului de a-și proteja datele și utilizatorii împotriva accesului neautorizat.

Sursa de stimul: Atacatorii externi sau interni.

Stimul: Tentativa de a accesa sau a modifica datele utilizatorului sau de a prelua controlul sistemului.

Mediu: Utilizarea de conexiuni nesigure.

Componentă: Sistemul web și baza de date.

Răspuns: Sistemul trebuie să verifice și să autentifice utilizatorii si să prevină accesul neautorizat.

Măsură de răspuns: Numărul de încercări nereușite de autentificare sau accesare.

Tactici: Autentificarea și autorizarea utilizatorilor, monitorizarea evenimentelor de securitate.

## Testability

Definiție: Capacitatea sistemului de a fi testat eficient pentru a verifica funcționarea corectă și respectarea cerințelor.

Sursa stimulului: Producatorul (eu) care testează sistemul.

Stimul: Testarea sistemului cu seturi de date și scenarii de testare.

Mediu: Condiții de testare controlate și configurate.

Artefact: Intregul sistem sau părți ale acestuia care sunt testate.

Răspuns: Furnizarea rezultatelor testelor și a erorilor identificate.

Măsură de răspuns: Procentul de cazuri de testare în care sistemul a trecut testele și a identificat cu succes erorile.

Tactice: Proiectare modulară, testare automată, simularea erorilor.

## Usability

Definiție: Capacitatea sistemului de a fi ușor de utilizat și intuitiv pentru utilizatorii finali.

Sursa stimulului: Utilizatorii finali care utilizează sistemul.

Stimul: Utilizarea sistemului în diferite scenarii de utilizare.

Mediu: Mediul de utilizare al sistemului.

Artefact: Interfața utilizatorului și elementele sale asociate.

Răspuns: Utilizarea ușoară și intuitivă a sistemului de către utilizatori.

Măsură de răspuns: Timpul necesar pentru ca utilizatorii să învețe să utilizeze sistemul.

Tactice: Proiectarea centrată pe utilizator.

# Design Constraints

Limbajul de programare: Python.

Dezvoltarea trebuie să respecte procesul Django.

Bibliotecile Django trebuie utilizate pentru a implementa funcționalitățile specifice ale aplicației.

Utilizarea framework-ului Django pentru arhitectura generală a aplicației.

Implementarea Factory Method și Observer Design Pattern.