Software Architecture Document

For NFS Infinity App

By David Rosenberg
U00063482
Contact e-mail: dcrsnbrg@memphis.edu

Submitted
November 9, 2017

Introduction

This document is intended to provide the reader with some high-level models of NFS Infinity, a Django-based network file server app being developed as a student software engineering project exercising agile software development methods. Three different models will be examined: client-server system architecture, the Django infrastructure, and NFS Infinity classes.

Client-Server System Architecture

NFS Infinity is a network file server app, which generally follows client-server architectural patterns across a network. In the case of NFS Infinity, a stand-alone server provides access to directory structure and file system information, which clients can call on across a network infrastructure. The architectural model is shown in figure 1:

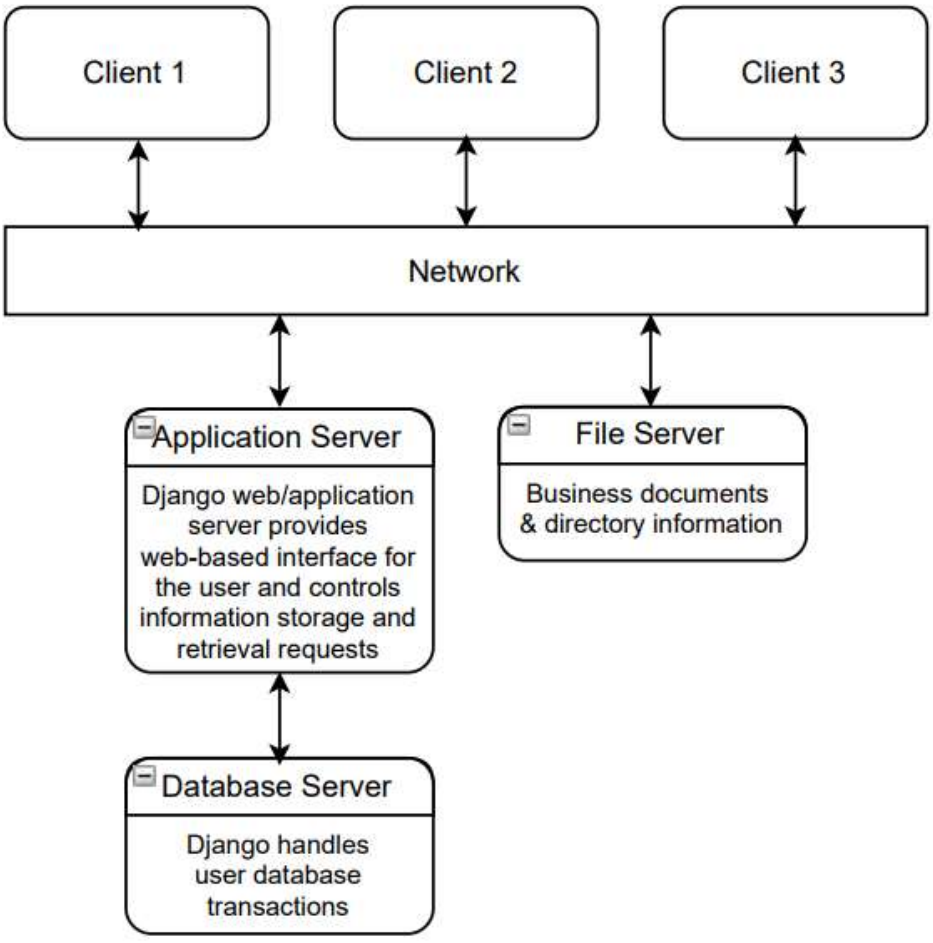

Figure 1: Client-server architecture of NFS Infinity

Django Infrastructure

Django is a Python web framework, which supports the MVC (model-view-controller) pattern, a web-based system where the user interface is implemented using a web browser. Django follows MVC with a slight variation: the controller component is handled by Django and the view is implemented by a template, an HTML (web) file with embedded DTL (Django Template Language). As described on tutorialspoint.com, figure 2 illustrates Django's MVC-MVT components and their interactions:

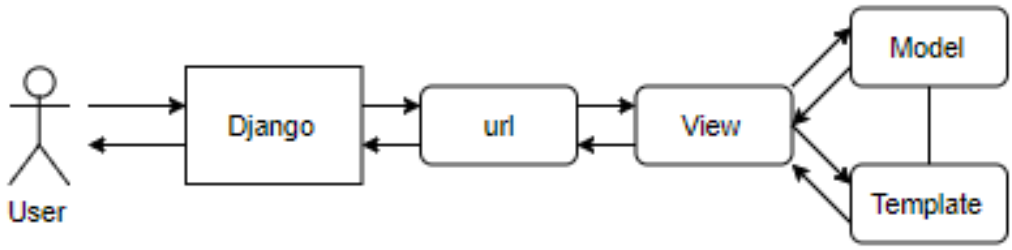

Figure 2: Django MVC-MVT pattern component interaction

NFS Infinity Classes

After gathering some basic requirements and developing some user stories, the initial design began to take form. It indicates three essential object classes: a user, an administrator, and a file system. The class diagram shown in figure 3 details some attributes and functions of each object as well as relationships between the objects:

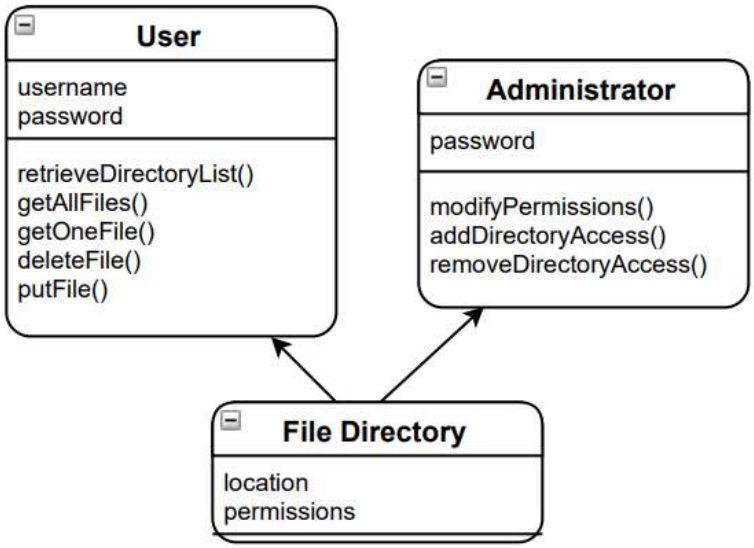

Figure 3: NFS Infinity class diagram