



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

UNIVERSITY OF PIRAEUS

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑ
ΠΜΣ "ΠΛΗΡΟΦΟΡΙΚΗ"

Ανάπτυξη Εφαρμογής για παραμετροποίηση δικτύου με το Django Framework

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΩΝ

Ιάσωνας Σιμώτας, Μαρία Μπαμπανέλου

Επιβλέπων: Πάνος Γκοτσιόπουλος
Καθηγητής ΠΑΠΕΙ

Επιβλέπων: Δουληγέρης Χρήστος
Καθηγητής ΠΑΠΕΙ

Αθήνα, Μήνας Έτος



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
UNIVERSITY OF PIRAEUS

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑ
ΠΜΣ "ΠΛΗΡΟΦΟΡΙΚΗ"
ΤΟΜΕΑΣ

Ανάπτυξη Εφαρμογής για παραμετροποίηση δικτύου με το Django Framework

ΔΙΠΛΩΜΑΤΙΚΗ
ΤΩΝ

Ιάσωνας Σιμωτας, Μαρία Μπαμπανέλου

Επιβλέπων: Πάνος Γκοτσιόπουλος
Καθηγητής ΠΑΠΕΙ
Επιβλέπων: Δουληγέρης Χρήστος
Καθηγητής ΠΑΠΕΙ

Εγκρίθηκε από την κάτωθι τριμελή επιτροπή την 1^η Ιανουαρίου 2019.

Όνομα Επώνυμο
Καθηγητής

Όνομα Επώνυμο
Καθηγητής

Όνομα Επώνυμο
Αναπληρωτής Καθηγητής

Ιάσοντας Σιμώτας, Μαρία Μπαμπανέλου
Πτυχιούχοι Μεταπτυχιακού ΠΜΣ Πληροφορικής

Copyright © Όνομα Επώνυμο, 2023

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ' ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πειραιά.

Περίληψη

Περίληψη διπλωματικής εργασίας.

Η παρούσα διπλωματική εργασία έχει σκοπό τη μελέτη και την ανάπτυξη λογισμικού για την παραμετροποίηση δικτυακών συσκευών με βασικό Framework το Django της Python. Η ιδέα για την υλοποίηση αυτής της εφαρμογής στάθηκε το υπόβαθρό μας στο software development ,στο χώρο του networking και με αφορμή μια εφαρμογή ενός μηχανικού της Cisco αποφασίσαμε να πάρουμε σαν βάση αυτήν και κάποιες ακόμα δημοσιεύσεις στο χώρο και να τις πάμε ένα βήμα παρακάτω . Σκοπός μας είναι να εισάγουμε τις έννοιες του microservices kubernetes και να μπορέσουμε να φτιάξουμε μια εφαρμογή η οποία περιλαμβάνει γνώσεις γύρω από διάφορους τομείς της Πληροφορικής. Εκτενής βιβλιογραφία θα παρουσιαστεί στο τέλος

Abstract

Diploma thesis abstract.

Keywords: Keyword 1, Keyword 2

Ευχαριστίες

Ευχαριστίες.

Περιεχόμενα

1	Εισαγωγή	13
1.1	Απαιτήσεις και προδιαγραφές	15
2	State of the art	17
2.1	Η επανάσταση στο web development	17
2.2	Frameworks και γιατί χρησιμοποιούνται	17
2.3	CI/CD pipeline	18
2.4	Τι είναι ο kubernetes	19

Κατάλογος Σχημάτων

2.1	Γενική αρχιτεκτονική του Django	17
-----	---	----

Κεφάλαιο 1

Εισαγωγή

Φαίνεται ότι ο χρόνος που σπαταλάνε οι μηχανικοί δικτύωσης όταν εισέρχονται σε εξοπλισμό δικτύωσης για να εισάγουν χειροκίνητα εντολές όπως για τη διαμόρφωση των συσκευών ή να εισέρχονται σε διακομιστές για τη χειροκίνητη ρύθμιση μία προς μία μια λίστα συσκευών/δικτύων(access lists) είναι πολύ μεγάλος, συνεπώς η εποχή που όλα αυτά γινόντουσαν χειροκίνητα φτάνει στο τέλος της. Όλο και περισσότεροι/περισσότερες εταιρείες προωθούν την αυτοματοποίηση καθώς βλέπουν ότι κάθε ώρα που επενδύεται στην αυτοματοποίηση μεταφράζεται σε πολλές ώρες εργασίας που εξοικονομούνται.

Η αυτοματοποίηση αυτών των εργασιών με κάποια καλοφτιαγμένη λογική προγραμματισμού επιτρέπει τη διαμόρφωση εκατοντάδων συσκευών μέσα σε λίγα λεπτά, απομακρύνει τη δυνατότητα των λανθασμένων ρυθμίσεων που προέρχονται από ανθρώπινο λάθος, επιτρέπει την καταγραφή των αλλαγών διαμόρφωσης και έχει το πλεονέκτημα ότι καθιστά τη διαμόρφωση τεχνικές λεπτομέρειες διαφανείς στον χρήστη που πρόκειται να ξεκινήσει τη διαδικασία αυτοματοποίησης. Για παράδειγμα, μια εταιρεία θα μπορούσε να αναθέσει υπερβολικά σε μια ομάδα λειτουργίας που δεν έχει τεχνικές γνώσεις δικτύωσης και απλά παρέχοντάς τους μια συγκεκριμένο σύνολο εισόδων θα μπορούσαν να διαμορφώσουν για Χ σημεία πρόσβασης σε Υ δίκτυα ένα συγκεκριμένο ΣΣΙΔ με τις επιθυμητές παραμέτρους. Οι δεδομένες εισοδοί θα μπορούσαν να εισαχθούν από αυτούς σε μια εφαρμογή ιστού και ο υποκείμενος προγραμματισμός κώδικας θα έκανε τα υπόλοιπα. Τελικά αυτό μεταφράζεται σε ένα πολύ γρήγορο και αξιόπιστο πλάνο κατά το οποίο η παραμετροποίηση και η διαμόρφωση ενός δικτύου δεν θα χρειάζεται να γίνει από τους μηχανικούς δικτύωσης χειροκίνητα.

Μπορούν να επενδύσουν συνεπώς αυτόν τον επιπλέον χρόνο σε άλλες εργασίες όπως ο σχεδιασμός και έτσι οι πιο χρονοβόρες διαδικασίες να αυτοματοποιηθούν. Αλλά η αυτοματοποίηση δεν είναι μόνο κάνει θαύματα όσον αφορά τη διαμόρφωση, είναι επίσης εξαιρετική για την παρακολούθηση της κατάστασης των δικτύων/συσκευών/θυρών, την απόκτηση πληροφοριών για την υγεία των ασύρματων δικτύων και κάθε άλλες πληροφορίες που μπορούν να λάβουν από της δυκτυακές συσκευές.

Είναι σημαντικό να ληφθεί υπόψη ότι οι επαναλαμβανόμενες καθημερινές/εβδομαδιαίες εργασίες που απαιτούν τη συλλογή πληροφοριών είναι εξαιρετικοί υποψήφιοι για αυτοματοποίηση. Με μια αυτοματοποίηση που αναζητά τα απαιτούμενα δεδομένα και κάνει κάποια επεξεργασία οι απαιτούμενες πληροφορίες μπορούν να ληφθούν γρήγορα και να παρουσιαστούν στους μηχανικό και τον/την απαλλάσσει από το να συνδέεται χειροκίνητα σε πολλές συσκευές, να ελέγχει ορισμένων γραμμών διαμόρφωσης, κ.λπ.

Η αυτοματοποίηση συσκευών χρησιμοποιείται εδώ και πολλά χρόνια για τη διαχείριση βλαβών ή την παρακολούθηση του επιπέδου υπηρεσιών, αλλά με τις αυξανόμενες επιχειρηματικές ανάγκες προκύπτουν νέες προκλήσεις και νέες ευκαιρίες. Μία από αυτές τις ευκαιρίες είναι η επένδυση των εταιρειών σε αυτοματισμούς δικτύων. Η ραγδαία ανάπτυξη των σύγχρονων δικτύων στις επιχειρήσεις μαζί με τις νέες τεχνολογίες, όπως το Διαδίκτυο των πραγμάτων (IOT) και το υπολογιστικό νέφος που βασίζονται επίσης στο δίκτυο, οδήγησαν στην ανάγκη ανάπτυξης της δικτυακής υποδομής με αποτέλεσμα την αύξηση του φόρτου εργασίας. απαιτήσεις για την παροχή, τη συντήρηση, την παρακολούθηση και τη διαχείριση από το δίκτυο.

Οι μέθοδοι που χρησιμοποιούσαν μέχρι σήμερα οι μηχανικοί δικτύων δεν ήταν μόνο χρονοβόρες αλλά και απαιτούνταν και γνώσεις σχετικά με ιδιόκτητα πρωτόκολλα και τεχνολογίες. Σε μια προσπάθεια να μειώσουν το κόστος και να δημιουργήσουν αποτελεσματικότητα οι μηχανικοί δικτύου ανέπτυξαν το Network automation ως τεχνικές αυτοματοποίησης για την αυτοματοποίηση επαναλαμβανόμενων καθημερινών εργασιών. Με την υποστήριξη σχεδόν όλων των μεγάλων εταιρειών δικτύωσης (όπως η Cisco) δημιουργήθηκε μια κοινότητα ανοιχτού κώδικα που είχε ως στόχο την υλοποίηση εφαρμογών αυτοματοποίησης κυρίως με τη χρήση τυποποιημένων διεπαφών (SSH, REST) και γενικών γλωσσών προγραμματισμού όπως η python. Με τη χρήση της Python και μιας συλλογής εννοιών και συναρτήσεων θα προσπαθήσουμε να φτιάξουμε μία εφαρμογή που συνδέει όλα τα παραπάνω.

Παράλληλα η επανάσταση που έφερε η εισαγωγή της λογικής των microservices στον κλαδο της Πληροφορικής μπορεί να καθιστήσει την εφαρμογή αυτή ακόμα πιο αξιόπιστη γιατί μπορεί να συμβάλει στο σχεδιασμό ενός συστήματος λογισμικού με μεγαλύτερη αξιοπιστία καθώς και να προσφέρει όλες εκείνες της θετικές προεκτάσεις χρήσης αυτών. Θα γίνει λοιπόν μια προσπάθεια εισαγωγής τεχνολογιών διαχείρισης και ανάπτυξης microservices όπως kubernetes και containers. Τα πλεονεκτήματα της χρήσης της αρχιτεκτονική Microservices είναι ότι προσφέρουν μεγαλύτερη ευελιξία μέσω της ανεξαρτησίας των υπηρεσιών, επιτρέποντας στους οργανισμούς να γίνουν πιο ευέλικτοι όσον αφορά τον τρόπο με τον οποίο προσφέρουν νέες επιχειρηματικές δυνατότητες ή ανταποκρίνονται στις μεταβαλλόμενες συνθήκες της αγοράς. Αναλυτική παρουσίαση αυτών θα γίνει σε επόμενο κεφάλαιο.

Σας συσκευές θα χρησιμοποιήσουμε αυτές της Cisco καθώς υπάρχουν ήδη βιβλιοθήκες οι οποίες υλοποιούν τα πρωτόκολλα επικοινωνίας και τις λειτουργίες που εμείς θέλουμε να υλοποιήσουμε. Η ανάπτυξη λογισμικού τέτοιων βιβλιοθηκών είναι

αντικείμενο μελέτης διπλωματικής εργασίας καθώς ξεφεύγει από τα πλαίσια μια μεταπτυχιακής διατριβής.

1.1 Απαιτήσεις και προδιαγραφές

- GNS3 VM ,Cisco Images και GNS3 περιβάλλον
- Vs Code development environment
- Virtual box ή οποιονδήποτε type B hypervisor

Κεφάλαιο 2

State of the art

2.1 Η επανάσταση στο web development

Στην αρχή, οι εφαρμογές ιστού δεν ήταν τίποτα περισσότερο από ένα μάτσο HTML, CSS και javascript που ήταν τοποθετημένα μαζί, συνδεδεμένα μεταξύ τους. Ένας καλός προγραμματιστής ήταν σε θέση να φτιάξει σπουδαίες εφαρμογές ιστού αν αυτός/αυτή είχε αρκετές δεξιότητες/γνώσεις.

Στην εποχή μας, εμφανίστηκαν τα frameworks και λαμβάνοντας υπόψη ότι δεν βελτιώνουν αυτό που τελικά βλέπει ο χρήστης και ο/η αλληλεπιδράσεις του με το frontend που τελικά είναι ο τελικός στόχος, τότε θα μπορούσε κανείς να αναρωτηθεί γιατί χρησιμοποιούνται ευρέως στις μέρες μας.

2.2 Frameworks και γιατί χρησιμοποιούνται

Στην τωρινή εποχή η ανάπτυξη λογισμικού είναι στενά συνδεδεμένη με τα frameworks. Η σύνδεση αυτή δεν είναι τυχαία καθώς η χρήση αυτών έχει κάνει τη ζωή των μηχανικών ανάπτυξης λογισμικού ευκολότερη. Θα εξηγήσουμε παρακάτω τους λόγους που συμβαίνει αυτό στο πλαίσιο κυρίως της δικιάς μας διπλωματικής εργασίας.



Σχήμα 2.1: Γενική αρχιτεκτονική του Django

- **Modularity** Καθώς η εφαρμογή μεγαλώνει, ο κώδικας πρέπει να είναι καλά δομημένος σε φακέλους και αρχεία ανάλογα με το τι κάνει ο κώδικας. Στο παρελθόν, οι μεγάλες εφαρμογές υπέφεραν όταν η εφαρμογή μεγάλωνε, υπήρχαν προβλήματα επεκτασιμότητας καθώς ο αριθμός των αρχείων javascript και CSS αυξανόταν ραγδαία και υπήρχαν πολύς επαναλαμβανόμενος κώδικας μεταξύ

των αρχείων. Με την παροχή μιας καθορισμένης δομής, ένα συγκεκριμένο κομμάτι κώδικα μπορεί να αναζητηθεί εύκολα. Αν πάρουμε ως παράδειγμα το πλαίσιο Django, το Django δομεί τον κώδικα σε ένα πολύ συγκεκριμένο τρόπο. Μέσα στο αρχείο `models.py` ορίζονται τα μοντέλα της βάσης δεδομένων. Αυτό συμβαίνει προκειμένου να μπορούν να γίνουν ερωτήματα στη βάση δεδομένων που δεν σχετίζονται με τη συγκεκριμένη βάση δεδομένων που χρησιμοποιείται στο έργο. Μέσα στο αρχείο `views.py` γίνεται η λογική για την ανάκτηση και την επεξεργασία των δεδομένων όταν το ζητάει ο χρήστης. Μέσα στο αρχείο `urls.py` υλοποιείται η δρομολόγηση της εφαρμογής. Μέσα στο φάκελο `templates` υπάρχουν όλα τα `.html` αρχεία στα οποία το αρχείο `views.py` στέλνει τα δεδομένα που λαμβάνει για να τα απεικονίσει, κ.λπ.

- Ταχύτερη ανάπτυξη Τα πλαίσια παρέχουν έτοιμες λειτουργίες, καλώντας απλώς τις ήδη ενσωματωμένες συναρτήσεις/μεθόδους. Διαβάζοντας απλώς την τεκμηρίωση του πλαισίου και μαθαίνοντας πώς να τις χρησιμοποιεί, ο προγραμματιστής μπορεί να ενσωματώσει λειτουργικότητες που διαφορετικά θα ήταν δύσκολο να υλοποιήσει και επίσης πολύ χρονοβόρες. Παραδείγματα περιλαμβάνουν λειτουργίες ελέγχου ταυτότητας, λειτουργίες διαχείρισης συνεδριών, λειτουργίες λειτουργίας βάσεων δεδομένων, λειτουργίες επικύρωσης φορμών και λειτουργίες για την παροχή ασφάλειας έναντι κακόβουλων επιτιθέμενων.
- Επέκταση κώδικα Τα περισσότερα frameworks επιτρέπουν την επέκταση κάποιου κομματιού κώδικα που θα χρησιμοποιηθεί σε πολλά άλλα αρχεία. Αυτό εξασφαλίζει ότι δεν υπάρχει επαναλαμβανόμενος κώδικας και οποιαδήποτε αλλαγή σε αυτόν τον κώδικα μεταφράζεται σε όλες τις περιπτώσεις που χρησιμοποιούν αυτόν τον κώδικα.
- Ευκολότερη αναγνωσιμότητα του κώδικα Δεδομένου ότι ο κώδικας χρησιμοποιεί καλά καθορισμένες τυποποιημένες συναρτήσεις και μια συγκεκριμένη δομή, είναι ευκολότερο να τον καταλάβει κάποιος που είναι νέος στο κώδικα αλλά γνωρίζει πώς λειτουργεί το πλαίσιο

2.3 CI/CD pipeline

Μόλις η εφαρμογή ιστού συνδεθεί με το απομακρυσμένο αποθετήριο, η τελευταία τάση στο στον κόσμο του DevOps είναι η υλοποίηση ενός αγωγού CI/CD, ο οποίος ουσιαστικά είναι μια αυτοματοποιημένη διαδικασία που ενεργοποιείται όταν νέος κώδικας δημοσιεύεται στο απομακρυσμένο αποθετήριο. Αυτή η διαδικασία ξεκινάει τη δημιουργία κώδικα, εκτελεί κάποιες δοκιμές και τέλος, αν όλα είναι εντάξει, αναπτύσσει αυτόματα τον κώδικα στην παραγωγή περιβάλλον. Με αυτόν τον τρόπο, οι προγραμματιστές μπορούν να διασφαλίσουν ότι τίποτα δεν θα χαλάσει στην παραγωγή και οι νέες λειτουργικότητες εξυπηρετούνται το συντομότερο δυνατό στους

πελάτη. Στην περίπτωσης μας τόσο η διπλωματική εργασία(latex) όσο και η εφαρμογή υλοποιήθηκαν με αυτή τη λογική.

2.4 Τι είναι ο kubernetes

Ο κυβερνήτης είναι ο διαχειριστής των με απλά λόγια ο διαχειριστής των containers.