



# Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών  
Τμήμα Μηχανικών Πληροφορικής

## Πτυχιακή Εργασία

«Ολοκληρωμένο Σύστημα Παροχής Τουριστικών  
Υπηρεσιών σε γλώσσα Python»

**Βασίλειος Βασιλάκης (ΑΜ: 3338)**

**Ευάγγελος Ξανθός (ΑΜ: 3211)**

**Επιβλέπων Καθηγητής: Νικόλαος Παπαδάκης**

## Ευχαριστίες

Θα θέλαμε να ευχαριστήσουμε τον κ. Νικόλαο Παπαδάκη για την ανάθεση της πτυχιακής εργασίας και για τη βοήθεια και καθοδήγησης που μας παρείχε κατά τη διάρκεια της εκπόνησής της.

Επίσης θα θέλαμε να ευχαριστήσουμε τις οικογένειες μας για τη στήριξή τους καθ' όλη τη διάρκεια των σπουδών μας.

## Abstract

The main goal of this thesis is to create a tourist guide. This tourist guide will provide the user with information about tourist attractions from the area he/she is going to visit, as well as info about hotels and rental offices and travel agencies. Through the tourist guide, the user can make reservations in whole or in part from the services presented to him and provided to the driver. We included all the above, sample points of interest in the areas of Heraklion and Chania.

## Σύνοψη

Κύριος στόχος της πτυχιακής αυτής εργασίας είναι η δημιουργία ενός τουριστικού οδηγού. Ο τουριστικός οδηγός αυτός θα παρέχει στο χρήστη πληροφορίες για αξιοθέατα τουριστικού ενδιαφέροντος της περιοχής που πρόκειται να επισκεφθεί, καθώς και για τα ξενοδοχεία και γραφεία ενοικιάσεων & ταξιδιωτικά γραφεία. Μέσω του τουριστικού οδηγού θα μπορεί ο χρήστης να κάνει κρατήσεις ολικώς ή μερικώς από τις υπηρεσίες που του παρουσιάζονται και παρέχονται στον οδηγό. Συμπεριλάβαμε στον οδηγό όλα τα παραπάνω, δειγματικά σημεία ενδιαφέροντος σε περιοχές Ηρακλείου και Χανίων.

## Πίνακας περιεχομένων

### Περιεχόμενα

Ευχαριστίες .....	2
Abstract .....	3
Σύνοψη .....	4
Πίνακας περιεχομένων .....	5
1. Εισαγωγή .....	7
1.1 Περίληψη.....	7
1.2 Κίνητρο για τη Διεξαγωγή της εργασίας .....	8
1.3 Σκοπός και Στόχοι Εργασίας .....	9
1.4 Δομή Εργασίας .....	9
2. Μεθοδολογία υλοποίησης.....	10
3. Σχέδιο δράσης/Προγραμματιστικά εργαλεία .....	11
3.1 Γλώσσα Python.....	11
3.1.1 Γενικές Πληροφορίες.....	11
3.1.2 Ιστορία της python .....	12
3.1.3 Εγκατάσταση Python.....	13
3.2 Django .....	15
3.2.1 Γενικές πληροφορίες.....	15
3.2.2 Εγκατάσταση Django .....	16
3.3 Βάση δεδομένων-SQL .....	19
3.3.1 Γενικές πληροφορίες.....	19
3.3.2 Ιστορία.....	19
3.4 PostgreSQL .....	20
3.4.1 Γενικές Πληροφορίες.....	20
3.4.2 Ιστορία.....	21
3.4.3 Εγκατάσταση PostgreSQL.....	22
3.5 Django &PostgreSQL .....	24
4. Κύριο μέρος Πτυχιακής .....	27
4.1 Ανάλυση προβλήματος .....	27
4.2 Δημιουργία App .....	27
4.3 Δημιουργία μοντέλων .....	28
4.4 Templates .....	32
4.4.1 Template bracket .....	32

4.4.2	Security token.....	33
4.4.3	Html, css & images .....	33
4.4.4	Template URLs.....	34
4.4.5	Template Images .....	35
4.5	Views .....	48
4.6	Σελίδα Admin .....	49
5.	Αποτελέσματα.....	53
5.1	Συμπεράσματα.....	53
5.2	Επεκτάσεις.....	53
	Βιβλιογραφία .....	54

## 1. Εισαγωγή

Ο τουρισμός είναι από τα σημαντικότερα στοιχεία που αναδεικνύει τη χώρα μας και αναπόσπαστο κομμάτι της ελληνικής οικονομίας. Με την ραγδαία εξέλιξη της τεχνολογίας των υπολογιστών και του διαδικτύου καθώς και μαζικότερη χρήση των τεχνολογικών συσκευών (PCs ,Smartphones etc.) και κοινωνικών δικτύων (“social media”) την τελευταία δεκαετία, οδηγεί το μέσο άτομο να αναζητεί οτιδήποτε θέλει στο διαδίκτυο.

Τα γεγονότα αυτά δεν θα μπορούσαν να αφήσουν ανεπηρέαστο τον τουρισμό, πλέον κάθε επιχείρηση ή φορέας τουρισμού στρέφεται σε οποιοδήποτε διαδικτυακό μέσο προκειμένου να προωθήσει τις υπηρεσίες του και από την άλλη κάθε χρήστης που θέλει να ταξιδέψει κάπου, αναζητά πρώτα πληροφορίες για την περιοχή (ιστορία, αξιοθέατα, εικόνες, τοπία κλπ.) αλλά και στην αναζήτηση τιμών και ποιότητας καταλυμάτων και σημείων εστίασης. Υπάρχει πληθώρα sites σε παγκόσμιο επίπεδο που προσφέρουν τέτοιες υπηρεσίες (tripadvisor.com, booking.com κλπ.), τα οποία είναι γενικού ενδιαφέροντος, αλλά προκειμένου να αναδειχθεί περισσότερο μια συγκεκριμένη περιοχή υπάρχει μια περαιτέρω ανάγκη, κι εδώ είναι που γίνεται απαραίτητη η χρησιμότητα ενός τουριστικού οδηγού της περιοχής αυτής.

### 1.1 Περίληψη

Σκοπός της πτυχιακής αυτής είναι η κατασκευή ενός ολοκληρωμένου συστήματος για την παροχή τουριστικών υπηρεσιών. Το σύστημα θα προσφέρει στον χρήστη πληροφορίες προκειμένου να κατατοπιστεί σχετικά με την για ψυχαγωγία του και τις διακοπές σε μια συγκεκριμένη περιοχή. Αναλυτικότερα παρέχονται πληροφορίες για:

1. Τα ξενοδοχεία της περιοχής (ονομασία, διεύθυνση, κατηγορία, τηλέφωνο, αριθμός & κατηγορία δωματίων)
2. Τα γραφεία ενοικιάσεων αυτοκινήτων (ονομασία, διεύθυνση, τηλέφωνο) καθώς και
  - i. Πληροφορία για το κάθε αυτοκίνητο που διαθέτει το κάθε γραφείο (αριθμό κυκλοφορίας, κυβικά, θέσεις επιβατών και κατηγορία στην οποία ανήκει). Επίσης παρέχεται πληροφορία όσον αφορά την χρέωση ανά μέρα (αν αυτή αλλάζει καθώς αυξάνονται οι μέρες ενοικιάσεων φαίνεται στο άθροισμα .όταν ο χρήστης κάνει booking).
  - ii. Πληροφορία για το κάθε μηχανάκι/μηχανή που διαθέτει το κάθε γραφείο(αριθμό κυκλοφορίας, κυβικά και κατηγορία). Πληροφορίες για την χρέωση.
3. Πληροφορίες για τα μουσεία (ονομασία, διεύθυνση, τιμή εισόδου)
4. Πληροφορίες για τα αξιοθέατα (ονομασία, διεύθυνση, τιμή εισόδου)
5. Πληροφορίες για τα δρομολόγια των αεροπλάνων και πλοίων για την άφιξη και αναχώρηση στο/από συγκεκριμένο μέρος. Θα περιλαμβάνει τιμές, ώρες και μέρες και εταιρία η οποία κάνει την αντίστοιχη πτήση / ακτοπλοϊκό δρομολόγιο.
6. Πληροφορίες για τα εκδρομικά γραφεία της περιοχής (ονομασία, διεύθυνση, τηλέφωνο). Για κάθε γραφείο θα περιέχονται πληροφορίες για

- i. Ποιες εκδρομές διοργανώνει (αφετηρία προορισμό και ώρες που γίνονται) και ποιες είναι οι τιμές.
- ii. Πακέτα προσφορών για πολλά άτομα ή πολλές εκδρομές.

7. Πληροφορία για τα συνεργαζόμενα ξενοδοχεία , εταιρίες μεταφορές (αεροπορικές, ακτοπλοϊκές) και γραφεία ενοικιάσεων . Σε αυτήν την περίπτωση μπορεί να υπάρχουν πακέτα προσφορών που περιλαμβάνουν και τα δύο.

Όσον αφορά τις κρατήσεις, ο τουριστικός οδηγός θα υποστηρίζει τις παρακάτω διεργασίες:

1. Θα επιτρέπονται online κρατήσεις καθώς και ακυρώσεις αυτών μέσω διαδικτύου.
2. Θα πρέπει να επιτρέπεται στον χρήστη να ζητάει αν μπορεί να κάνει κράτηση η οποία θα περιλαμβάνει
  - i. δωμάτια (π.χ. 1 δίκλινα, 3 μονόκλινα κτλ.) και κατηγορία και περιοχή ξενοδοχείου
  - ii. κράτηση κάποιων μεταφορικών (αυτοκινήτων ή μηχανών) για κάποιο χρονικό διάστημα.
3. Το σύστημα θα κάνει την κράτηση μόνο αν όλα όσα ζητάει ο χρήστης είναι διαθέσιμα.
4. Επίσης ο χρήστης θα έχει δικαίωμα να κάνει κράτηση για άφιξη και αναχώρηση, οπότε η κράτηση θα γίνεται μόνο αν όλα όσα ζητάει ο χρήστης είναι διαθέσιμα. Δεν είναι απαραίτητο μια κράτηση να περιλαμβάνει όλα τα παραπάνω.

Επιπροσθέτως το σύστημα θα επιτρέπει στον χρήστη να κάνει αναζήτηση των πιο πάνω και θα του επιστρέφει τις εναλλακτικές λύσεις με τις αντίστοιχες χρεώσεις. Δεν είναι απαραίτητο μια αναζήτηση να περιέχει όλα τα παραπάνω. Η αναζήτηση μπορεί να περιλαμβάνει και εκδρομές.

Αξίζει να αναφέρουμε ότι ο τουριστικός οδηγός είναι ο μεσολαβητής και υπεύθυνος για οτιδήποτε αφορά τις κρατήσεις μεταξύ του χρήστη και των τουριστικών γραφείων/ξενοδοχείων/γραφείων ενοικιάσεων, οπότε αφού δε «μιλάνε» τα δύο άκρα απ' ευθείας, πολλές πληροφορίες από τα παραπάνω είναι περιττές για να φαίνονται στον απλό χρήστη, οπότε απλά τις κρατάει ο διαχειριστής στη βάση του και δεν απεικονίζονται στο site.

### 1.2 Κίνητρο για τη Διεξαγωγή της εργασίας

Γνωρίζοντας όλοι ότι στην Ελλάδα και ειδικά στην Κρήτη ο τουρισμός είναι ιδιαίτερα σημαντικός τομέας στην οικονομία και κατά συνέπεια στις ζωές των ανθρώπων που ασχολούνται ή εξαρτώνται από αυτό. Ως φοιτητές Μηχανικών Πληροφορικής στο Ηράκλειο αλλά και ως ντόπιοι κάτοικοι στην ευρύτερη περιοχή παρατηρούμε στην καθημερινότητά μας αλλά και στο γύρω μας περιβάλλον ότι μεγάλη μερίδα των ανθρώπων άμεσα ή έμμεσα ασχολείται ή συναναστρέφεται με τον τουρισμό. Είτε σαν μονάδες ως εργαζόμενοι/ανθρώπινο δυναμικό σ' αυτές, είτε σαν επιχειρήσεις σ' αυτόν τον τομέα ή εξαρτώμενες από αυτόν. Οπότε θεωρήσαμε καλή ιδέα το αντικείμενο που θα ασχοληθούμε στην πτυχιακή μας να είναι ο τουρισμός, μ' αυτόν τον τρόπο συνδυάζουμε τις γνώσεις μας στον τομέα τις πληροφορικής σε συνδυασμό με ένα από τα πιο κύρια αντικείμενα



ενασχόλησης στον τόπο μας, έτσι ώστε αν βγει κανένα καλό αποτέλεσμα από την εργασία αυτήν να είναι πιο εύκολο να κάνουμε άνοιγμα και στην αγορά εργασίας του τουρισμού.

### 1.3 Σκοπός και Στόχοι Εργασίας

Η εργασία αυτή έχει ως πρωταρχικό στόχο την απόκτηση επιπλέον γνώσεων και μιας αρχικής εμπειρίας γύρω από το web development, πράγμα που θα μπορούσε να επιτευχθεί μέσω της μελέτης και της έρευνας όλων των απαραίτητων μεθόδων και εργαλείων που θα χρησιμοποιήσουμε και ασφαλώς η πλήρης κατανόησή τους προκειμένου να μπορεί να υλοποιηθεί η εργασία. Επιπλέον στόχοι, όσον αφορά το θέμα που επιλέξαμε να εστιάσουμε, τον τουρισμό, είναι μέσω του οδηγού αυτού, να παρέχουμε στο μέσο χρήστη/επισκέπτη να μπορεί να προηγηθεί με ευκολία και να μπορεί να μάθει πληροφορίες σχετικά με τη διαμονή του και την ψυχαγωγία του στο μέρος που πρόκειται να επισκεφθεί, επίσης ο οδηγός θα πρέπει να τον κάνει να προσελκυστεί και να επιλέξει αυτό το μέρος, όπως πχ με την επίδειξη πακέτων προσφορών κ.α.

### 1.4 Δομή Εργασίας

Στο πρώτο κεφάλαιο γίνεται μια περιληπτική αναφορά στο περιεχόμενο της πτυχιακής εργασίας αυτής, το κίνητρο που μας οδήγησε να ασχοληθούμε με το θέμα αυτό, καθώς και τους στόχους και τους σκοπούς μας.

Στο δεύτερο κεφάλαιο γίνεται αναφορά και στις μεθόδους που χρησιμοποιήσαμε για να αναπτύξουμε την πτυχιακή εργασία μας.

Στο τρίτο κεφάλαιο αναφέρονται αναλυτικά όλα τα προγραμματιστικά εργαλεία που χρησιμοποιήσαμε, και εκτενή επεξήγηση ανάλυση του καθενός. Επιπλέον αναλύονται και επεξηγούνται οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν, τα απαραίτητα καθώς και τα βοηθητικά προγράμματα και εργαλεία λογισμικού που χρειαστήκαμε και εγκαταστήσαμε.

Στο τέταρτο κεφάλαιο έχουμε το κύριο μέρος της πτυχιακής εργασίας όπου αρχικά γίνεται η ανάλυση του προβλήματος. Στη συνέχεια γίνεται αναφορά στο σχεδιασμό υλοποίησης, δηλαδή επεξηγούμε τα βήματα που κάναμε για να φτάσουμε στο επιθυμητό αποτέλεσμα, (επεξήγηση αρχείων κώδικα, στήσιμο βάσης δεδομένων κλπ.) Και εν τέλει παρουσιάζεται η τελική εργασία επεξηγώντας κάθε κομμάτι και λειτουργία του site, από την πλευρά διαχειριστή και απλού χρήστη/επισκέπτη.

Στο πέμπτο κεφάλαιο κάνουμε μια σύνοψη για το τι αποτελέσματα και οφέλη αποκομίζουμε με το πέρας της πτυχιακής αυτής, καθώς και συμπεράσματα και πιθανές μελλοντικές βλέψεις και χρήση αυτής.

## 2. Μεθοδολογία υλοποίησης

Για τη δημιουργία της ιστοσελίδας χρησιμοποιήθηκε το Django Framework το οποίο είναι ένα framework γραμμένο σε γλώσσα python για την κατασκευή ιστοσελίδων σε σύντομο χρονικό περιθώριο. Η χρήση του framework γίνεται κυρίως με python και για την παρουσίαση των δεδομένων στο χρήστη χρησιμοποιούνται επιπλέον html και css.

Για την βάση δεδομένων χρησιμοποιήθηκε η PostgreSQL όπου θεωρείτε η πιο ανεπτυγμένη βάση δεδομένων ελεύθερου λογισμικού σε σχέση με τη MySQL που επιτρέπει πολλά λογικά λάθη όπως την πράξη της διαίρεσης μηδέν δια μηδέν που είναι μια πράξη που δεν μπορεί να γίνει και άλλα παρόμοια λογικά λάθη.. Η εφαρμογή που χρησιμοποιήσαμε είναι το pgAdmin 4 όπως θα δούμε και παρακάτω..

Τέλος όλο το project γράφτηκε πάνω στο ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) της Microsoft Visual Studio 2015 το οποίο είχαμε στην διάθεση δωρεάν μας και στην πλήρη έκδοση του από την σαν φοιτητές του Τ.Ε.Ι Κρήτης από την ιστοσελίδα [MSDNAA](#) της Microsoft σε συνεργασία με το Τ.Ε.Ι Κρήτης. Η χρήση του βοήθησε πολύ στο να απλουστεύσει και να κάνει πιο γρήγορη την ανάπτυξη της ιστοσελίδας μιας και παρέχει και εγκαθιστά πολύ εύκολα και γρήγορα μέσα από το ίδιο οποιοδήποτε επιπλέον πρόγραμμα χρειαστήκαμε.

### 3. Σχέδιο δράσης/Προγραμματιστικά εργαλεία

#### 3.1 Γλώσσα Python



##### 3.1.1 Γενικές Πληροφορίες

Η Python είναι μια υψηλού επιπέδου γλώσσα προγραμματισμού. Δημιουργήθηκε από τον Guido van Rossum το 1990 και κυκλοφόρησε για πρώτη φορά το 1991. Κύριος στόχος καθώς και φιλοσοφία της είναι να είναι όσο το δυνατόν αναγνωρίσιμος ο κώδικάς της καθώς και ο τρόπος σύνταξής της, έτσι ώστε να καθιστά τη χρήση της με ευκολία απέναντι στους προγραμματιστές και να τους επιτρέπει να την εκφράσουν έννοιες σε λιγότερες γραμμές κώδικα απ'ότι θα ήταν δυνατόν σε γλώσσες όπως η C++ ή η Java . Διακρίνεται λοιπόν για την ευκολία της στην ανάγνωση γι' αυτό και είναι από τις πρώτες επιλογές που προτείνεται για αρχάριους.

Έχει μια μεγάλη και ολοκληρωμένη τυποποιημένη βιβλιοθήκη. Η Python είναι μια ερμηνευμένη γλώσσα. Οι ερμηνευμένες γλώσσες δεν χρειάζεται να συνταχθούν για να τρέξουν. Ένα πρόγραμμα που ονομάζεται διερμηνέας θα εκτελέσει κώδικα Python σε οποιοδήποτε υπολογιστή μπορεί να τρέξει στον εαυτό του. Οι διερμηνείς της Python είναι διαθέσιμοι για πολλά λειτουργικά συστήματα, επιτρέποντας στον κώδικα Python να λειτουργεί σε μια μεγάλη γκάμα συστημάτων. Η Python αναπτύσσεται ως λογισμικό ανοιχτού κώδικα (open source) και η διαχείρισή της γίνεται από τον μη κερδοσκοπικό οργανισμό Python Software Foundation.

Η Python διαθέτει ένα σύστημα δυναμικού τύπου και αυτόματη διαχείριση μνήμης και υποστηρίζει πολλά παραδείγματα προγραμματισμού, συμπεριλαμβανομένων των αντικειμενοστραφών, λειτουργικών προγραμμάτων κ.α.

Η συνηθισμένη βιβλιοθήκη αποτελείται από πολλές λειτουργίες που συνοδεύουν την Python όταν είναι εγκατεστημένη. Στο Διαδίκτυο υπάρχουν πολλές άλλες βιβλιοθήκες που επιτρέπουν στη γλώσσα Python να κάνει περισσότερα πράγματα. Αυτές οι βιβλιοθήκες την καθιστούν μια ισχυρή γλώσσα. μπορεί να κάνει πολλά διαφορετικά πράγματα.

Μερικά πράγματα για τα οποία χρησιμοποιείται συχνά η Python είναι:

Web development

Προγραμματισμός δικτύων

Προγραμματισμός παιχνιδιών

Desktop GUIs

Επιστημονικός προγραμματισμός

### 3.1.2 Ιστορία της python

Αρχικά, η Python ήταν γλώσσα σεναρίων που χρησιμοποιούνταν στο λειτουργικό σύστημα Amoeba, ικανή και για κλήσεις συστήματος.

Η Python σχεδιάστηκε στα τέλη της δεκαετίας του 1980 και η εφαρμογή της άρχισε τον Δεκέμβριο του 1989 από τον Guido van Rossum στο Centrum Wiskunde & Informatica της Ολλανδίας.

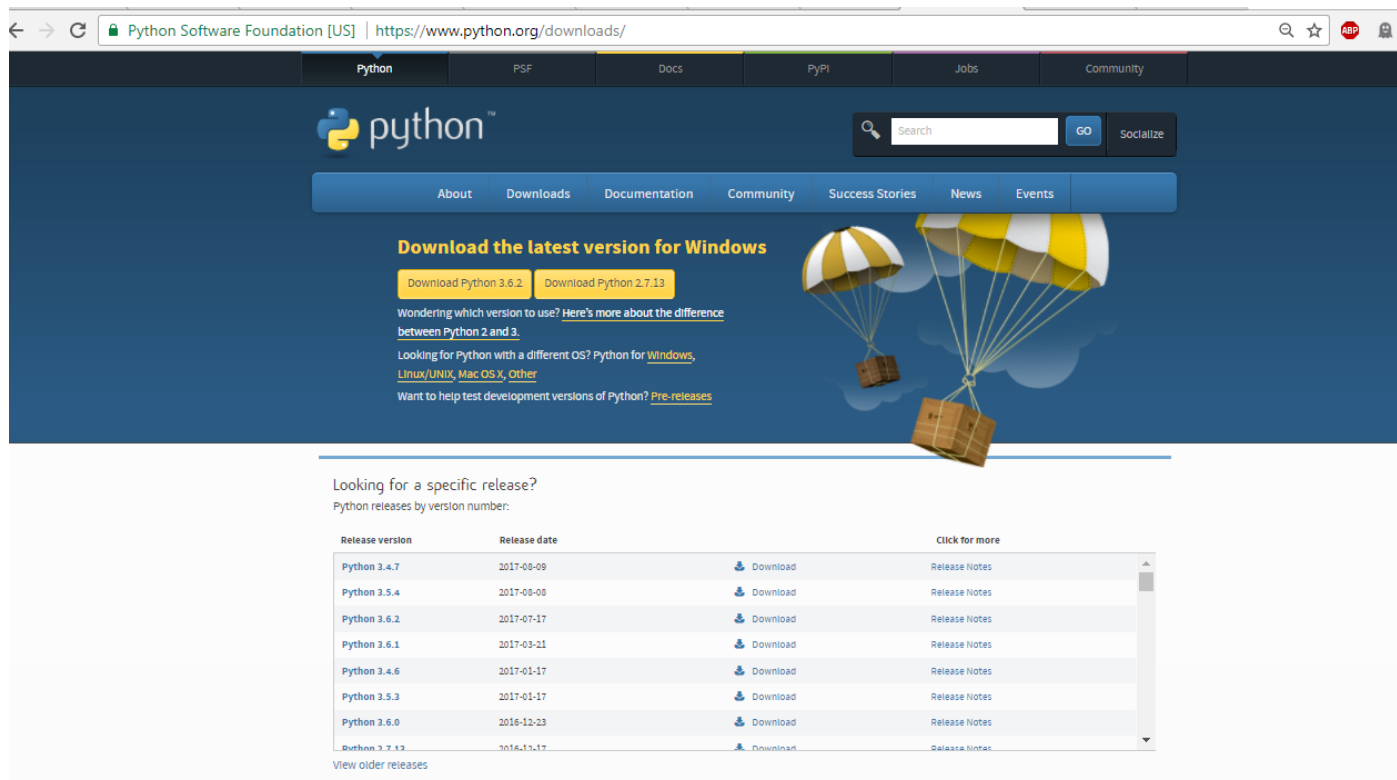
Η Python 2.0 κυκλοφόρησε τον Οκτώβριο του 2000 και είχε πολλά σημαντικά νέα χαρακτηριστικά, συμπεριλαμβανομένου ενός συλλέκτη απορριμμάτων και υποστήριξη της Unicode. Το Δεκέμβριο του 2008 κυκλοφόρησε η έκδοση 3.0 (γνωστή και ως py3k ή python 3000). Πολλά από τα καινούργια χαρακτηριστικά αυτής της έκδοσης έχουν μεταφερθεί στις εκδόσεις 2.6 και 2.7 που είναι προς τα πίσω συμβατές (backward compatible).

Η Python 3 είναι ιστορικά η πρώτη γλώσσα προγραμματισμού που σπάει την προς τα πίσω συμβατότητα με προηγούμενες εκδόσεις ώστε να διορθωθούν κάποια λάθη που υπήρχαν σε προγενέστερες εκδόσεις και να καταστεί ακόμα πιο σαφής ο απλός τρόπος με τον οποίο μπορούν να γίνουν κάποια πράγματα.

Η εκδόσεις της Python 2 με τελευταία πρόσφατη την Python 2.7 ήταν σχεδιασμένη να εγκαταληφθεί σιγά σιγά, αρχικά ορίστηκε ως ημερομηνία το 2015, όμως στην πορεία αναβλήθηκε για το 2020 από την ανησυχία ότι ένα μεγάλο μέρος του υπάρχοντος κώδικα δεν θα μπορεί εύκολα να μεταφερθεί προς τα εμπρός στην Python 3.

### 3.1.3 Εγκατάσταση Python

Η εγκατάσταση του είναι απλή, απλά επισκεπτόμαστε το επίσημο site της python για να την κατεβάσουμε (<https://www.python.org>) και επιλέγουμε τον κατάλληλο installer, τον οποίο θα κατεβάσουμε και θα εγκαταστήσουμε στον υπολογιστή μας ακολουθώντας τα βήματα. Θα πρέπει να επιλέξουμε ανάμεσα στις εκδόσεις της Python 2 και της 3. Το site μας παραπέμπει σε ένα σύνδεσμο προκειμένου να διευκολύνει το χρήστη να αποφασίσει ποια από τις δύο να επιλέξει και να μάθει τις διαφορές, αλλά η τελική απόφαση είναι στην κρίση του χρήστη για το ποια θα θεωρήσει καταλληλότερη για την εξυπηρέτησή του. Εμείς θεωρήσαμε καταλληλότερη την python 3. Κάνοντας scroll down μας δίνεται η δυνατότητα επίσης να επιλέξουμε να κατεβάσουμε κάποια συγκεκριμένη έκδοση απ' αυτές που κυκλοφόρησαν στο παρελθόν, πέρα από την προτεινόμενη τελευταία.



Παραπάνω φαίνεται μια ενδεικτική εικόνα από την ιστοσελίδα <https://www.python.org/downloads/> από όπου κατεβάσαμε το αρχείο με το οποίο θα γίνει η εγκατάσταση της Python..

Ανοίγοντας το αρχείο με την έκδοση που επιλέξαμε να χρησιμοποιήσουμε για να προχωρήσουμε στην υλοποίηση της εργασίας μας επιλέγουμε και το "κουτάκι"

Add python 3.5 to PATH όπως φαίνεται στην παρακάτω εικόνα.. Αυτό το κάνουμε για να "περάσει" η εγκατάσταση της Python στο αρχείο των Windows Path και έτσι θα έχουμε την όσον δυνατόν καλύτερη λειτουργία της χωρίς μελλοντικά σφάλματα όταν τρέχουμε ένα πρόγραμμα γραμμένο στην γλώσσα Python.. Μετά από αυτήν την επιλογή προχωράμε κανονικά στην εγκατάσταση της..



## 3.2 Django



### 3.2.1 Γενικές πληροφορίες

Το Django είναι ένα ελεύθερο και ανοιχτού κώδικα framework, γραμμένο σε γλώσσα Python το οποίο χρησιμοποιεί το model-view-template μοντέλο (MVT). Το Django δημιουργήθηκε το φθινόπωρο του 2003, όταν οι προγραμματιστές διαδικτύου στην εφημερίδα Lawrence Journal-World, Adrian Holovaty και Simon Willison, άρχισαν να χρησιμοποιούν την Python για την κατασκευή εφαρμογών. Δημοσιεύθηκε δημοσίως με άδεια BSD τον Ιούλιο του 2005.

Τον Ιούνιο του 2008, ανακοινώθηκε ότι ένα νεοσύστατο Django Software Foundation (DSF) θα διατηρούσε το Django στο μέλλον. Η απόφαση για την δημιουργία της ιστοσελίδας με το Django έγινε για τους ίδιους λόγους που αναφέρονται στην επίσημη ιστοσελίδα του:

#### Είναι πολύ γρήγορο

Έχει σχεδιαστεί ώστε να βοηθάει τους σχεδιαστές να υλοποιούν την ιδέα της εφαρμογής όσο το δυνατόν πιο γρήγορα.

#### Πλήρες εξοπλισμένο

Έχει πάρα πολλές επεκτάσεις που μπορούν να χρησιμοποιηθούν για απλές διαδικτυακές ανάγκες. Παρέχει το Site administration, το authentication του χρήστη καθώς και πολλές άλλες διεργασίες.

#### Πάρα πολύ ασφαλές

Παίρνοντας την ασφάλεια στα σοβαρά παρέχει έναν ασφαλή τρόπο διαχείρισης τους λογαριασμούς και τους κωδικούς των χρηστών. Μπορεί να αποφύγει σύνθετες λάθη ασφάλειας όπως sql injection, cross-site scripting και αλλά.

## Επεκτάσιμο

Μερικά από τις πιο πολυσύχναστες ιστοσελίδες χρησιμοποιούν το Django για τη δυνατότητα που έχει γρήγορα κι ευέλικτα να μεγαλώνει για να καλύψει τις ανάγκες κίνησης.

## Εξαιρετικά ευέλικτο

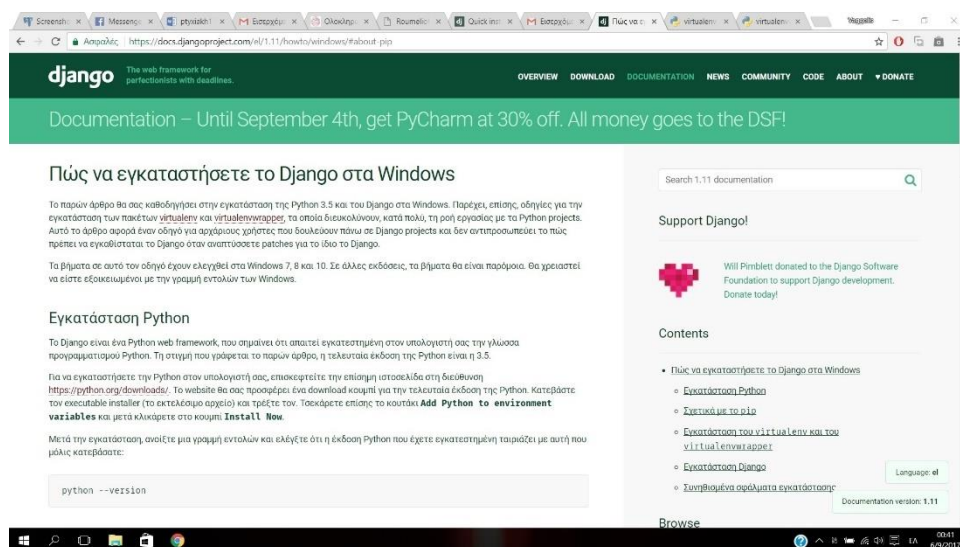
Εταιρίες, οργανισμοί και κυβερνήσεις έχουν χρησιμοποιήσει το Django για να δημιουργήσουν από συστήματα ελέγχου πληροφορίας και κοινωνικά δίκτυα μέχρι πλατφόρμες για επιστημονικούς υπολογισμούς.

Ο πρωταρχικός στόχος του Django είναι να διευκολύνει τη δημιουργία σύνθετων ιστότοπων που βασίζονται σε βάσεις δεδομένων. Το Django εστιάζει στην επαναχρησιμοποίηση και την συνδεσιμότητα των επι μέρους κομματιών, την ραγδαία ανάπτυξη και την αρχή της μη-επαναληψιμότητας. Η Python χρησιμοποιείται καθ' όλη τη διάρκεια, ακόμη και για αρχεία ρυθμίσεων και μοντέλα δεδομένων. Το Django παρέχει επίσης μια προαιρετική διεπαφή, δημιουργίας, ανάγνωσης, ενημέρωσης και διαγραφής που παράγεται δυναμικά μέσω ενδοσκόπησης και διαμορφώνεται μέσω μοντέλων admin.

### 3.2.2 Εγκατάσταση Django

Η εγκατάστασή του στο visual studio έγινε μαζί με το visual studio μιας και ήταν στις επιλογές που δίνει στην αρχή το visual studio. Το μόνο που χρειάστηκε να κάνουμε μέσω από το Django project στο visual studio ήταν η εγκατάσταση επεκτάσεων.

Η εγκατάστασή μπορεί να γίνει και από την γραμμή εντολών ακολουθώντας τον παρακάτω οδηγό στην ιστοσελίδα <https://docs.djangoproject.com/el/1.11/howto/windows/> εκτελώντας τις αντίστοιχες εντολές για την εγκατάσταση του κάθε προσθέτου που χρειάζεται για το Django



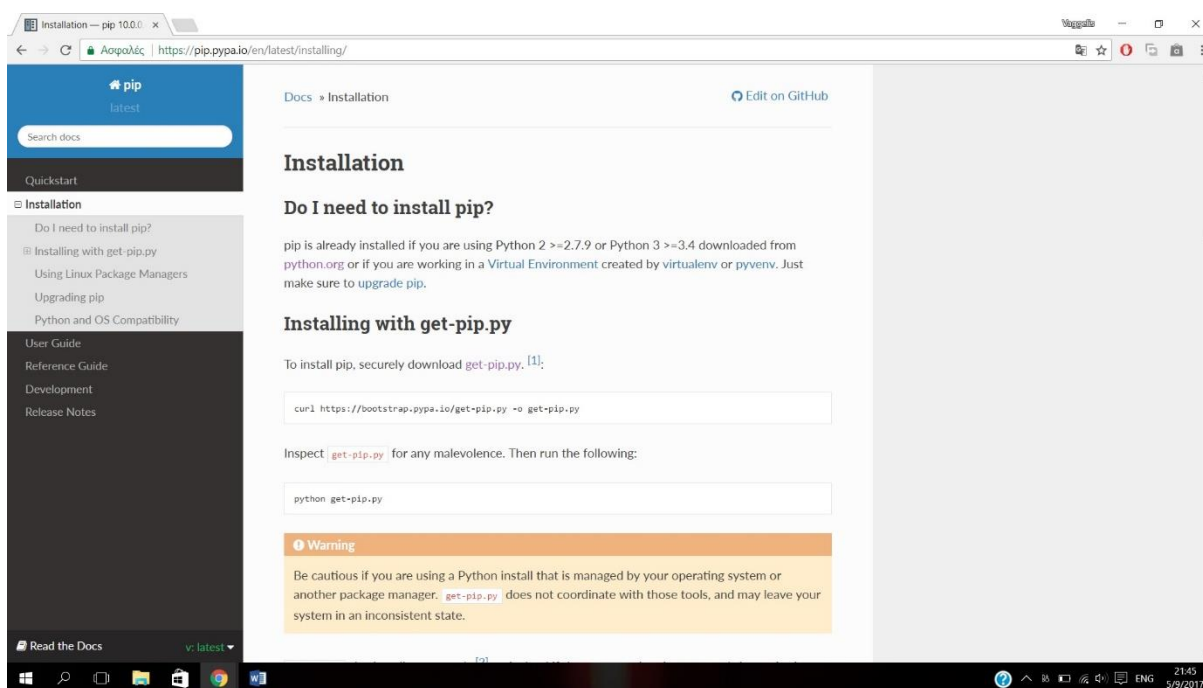


Από το περιβάλλον διαχείρισης που προσφέρει το visual studio με τη δημιουργία ενός Django project κατεβάσαμε την έκδοση της Python που επιθυμούμε. Στη συνέχεια εγκαταστήσαμε εκεί τις επεκτάσεις που χρειάστηκαν. Αυτές είναι:

## Pip

Το pip είναι ένα σύστημα διαχείρισης πακέτων που χρησιμοποιείται για την εγκατάσταση και τη διαχείριση πακέτων λογισμικού γραμμένα σε Python.

Το σύστημα διαχείρισης πακέτων pip μπορούμε να το εγκαταστήσουμε από την γραμμή εντολών κατεβάζοντας πρώτα το αρχείο get-pip.py από την ιστοσελίδα <https://pip.pypa.io/en/stable/installing/> και “τρέχοντας” μετά την εντολή python get-pip.py στην γραμμή εντολών



## Psycopg2

Το Psycopg είναι ο πιο δημοφιλής “adapter” βάσης δεδομένων PostgreSQL για τη γλώσσα προγραμματισμού Python . Τα κύρια χαρακτηριστικά του είναι η πλήρης εφαρμογή της προδιαγραφής Python DB API 2.0 και η ασφάλεια του “νήματος” (μερικά “νήματα” μπορούν να μοιράζονται την ίδια σύνδεση). Έχει σχεδιαστεί για εφαρμογές με πολλαπλές σπείρες που δημιουργούν και καταστρέφουν πολλούς δρομείς και κάνουν μεγάλο αριθμό ταυτόχρονων INSERTS ή UPDATES

Η εγκατάσταση του μπορεί να γίνει ακολουθώντας τις οδηγίες της ιστοσελίδας <https://pypi.python.org/pypi/psycopg2#downloads> από την γραμμή εντολών είτε κατεβάζοντας και τρέχοντας το αρχείο που μας παρέχεται από την ιστοσελίδα

The screenshot shows the PyPI page for the package `psycopg2` version `2.7.3.1`. The page title is `psycopg2 2.7.3.1` and the subtitle is `psycopg2 - Python-PostgreSQL Database Adapter`. The main content area contains the following text:

Psycopg is the most popular PostgreSQL database adapter for the Python programming language. Its main features are the complete implementation of the Python DB API 2.0 specification and the thread safety (several threads can share the same connection). It was designed for heavily multi-threaded applications that create and destroy lots of cursors and make a large number of concurrent "INSERT"s or "UPDATE"s.

Psycopg 2 is mostly implemented in C as a libpq wrapper, resulting in being both efficient and secure. It features client-side and server-side cursors, asynchronous communication and notifications, "COPY TO/COPY FROM" support. Many Python types are supported out-of-the-box and adapted to matching PostgreSQL data types; adaptation can be extended and customized thanks to a flexible objects adaptation system.

Psycopg 2 is both Unicode and Python 3 friendly.

**Documentation**

Documentation is included in the `doc` directory and is available online.

**Installation**

If your `pip` version supports wheel packages it should be possible to install a binary version of Psycopg including all the dependencies from PyPI. Just run:

```
$ pip install -U pip # make sure your pip is up-to-date
$ pip install psycopg2
```

If you want to build Psycopg from source you will need some prerequisites (a C compiler, development packages): please check the install and the faq documents in the `doc` dir for the details.

For any other resource (source code repository, bug tracker, mailing list) please check the project homepage.

**Linux/OSX:** [build](#) [passing](#)

**Windows:** [build](#) [passing](#)

The sidebar on the right contains a "Not Logged In" section with links for "Login", "Register", "Lost Login?", "Login with OpenID", and "Login with Google". Below this is a "Status" section with the text "Nothing to report".

### 3.3 Βάση δεδομένων-SQL

#### 3.3.1 Γενικές πληροφορίες

Η SQL (Structured Query Language) είναι μια γλώσσα που χρησιμοποιείται για την προβολή ή την αλλαγή δεδομένων σε βάσεις δεδομένων. Οι προτάσεις που χρησιμοποιούνται σε αυτή τη γλώσσα ονομάζονται SQL queries. Μπορούμε να χρησιμοποιήσουμε τα SQL Queries για να προβάλλουμε, να προσθέσουμε, να αφαιρέσουμε δεδομένα ή να δούμε δεδομένα στη βάση δεδομένων.

Είναι μία γλώσσα υπολογιστών στις βάσεις δεδομένων, η οποία αρχικώς είχε βασιστεί στη σχεσιακή άλγεβρα και στο σχεσιακός λογισμό πλειάδων. Σχεδιάστηκε για τη διαχείριση δεδομένων, σε ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (Relational Database Management System, RDBMS). Η γλώσσα περιλαμβάνει δυνατότητες ανάκτησης και ενημέρωσης δεδομένων, δημιουργίας και τροποποίησης σχημάτων και σχεσιακών πινάκων, αλλά και ελέγχου πρόσβασης στα δεδομένα. δεδομένων.

#### 3.3.2 Ιστορία

Το 1970 ο Δρ E.F. Codd δημοσίευσε ένα άρθρο με τίτλο «A Relational Model of Data for Large Shared Data Banks,», που περιγράφει ένα μοντέλο για την αποθήκευση και τον χειρισμό δεδομένων χρησιμοποιώντας πίνακες. Λίγο μετά τη δημοσίευση του άρθρου του Codd, η IBM άρχισε να εργάζεται για τη δημιουργία μιας σχεσιακής βάσης δεδομένων. Πρόσωπα-κλειδιά για την πρωταρχική ανάπτυξη της γλώσσας αυτής ήταν οι Donald D. Chamberlin και Raymond F. Boyce, οι οποίοι την ανέπτυξαν στις αρχές του '70 για λογαριασμό της IBM. Μεταξύ του 1979 και του 1982, η Oracle (η τότε Relational Software, Inc.), η Relational Technology, Inc. (που αργότερα αποκτήθηκε από την Computer Associates) και η IBM έβαλαν όλες τις εμπορικές σχεσιακές βάσεις δεδομένων και μέχρι το 1986 όλοι χρησιμοποιούσαν SQL ως γλώσσα επερωτήσεων (σ.σ. query language) .

Το 1986, το Αμερικανικό Εθνικό Ινστιτούτο Προτύπων (ANSI) τυποποίησε την SQL. Το πρότυπο αυτό ενημερώθηκε το 1989, το 1992 (που ονομάζεται SQL2), το 1999 (ονομάζεται SQL3), το 2003 (ονομάζεται SQL 2003), το 2006 (ονομάζεται SQL 2006) και το 2008 (ονομάζεται SQL 2008). Η τυπική SQL ονομάζεται μερικές φορές ANSI SQL. Όλες οι μεγάλες σχεσιακές βάσεις δεδομένων υποστηρίζουν αυτό το πρότυπο αλλά το καθένα έχει τις δικές του ιδιότητες επεκτάσεις.

Η SQL στην πορεία υιοθετήθηκε και αναπτύχθηκε από διάφορες άλλες εταιρίες πληροφορικής, τροποποιήθηκε και εξελίχθηκε σε διάφορες μορφές, ανάλογα τις απαιτήσεις, τους σκοπούς και τις ανάγκες κάθε εταιρίας. Οι διάφορες μορφές που δημιουργήθηκαν είναι οι επεκτάσεις τις standard SQL.

Οι επεκτάσεις της Standard SQL προσθέτουν λειτουργική γλώσσα προγραμματισμού, όπως είναι οι δομές ελέγχου ροής. Εκτός από τις τυπικές επεκτάσεις SQL και τις ιδιόκτητες επεκτάσεις SQL, η διαδικαστική (procedural) και αντικειμενοστραφή (object-oriented) προγραμματιστική δυνατότητα είναι διαθέσιμη σε πολλές πλατφόρμες SQL μέσω της ενσωμάτωσης του Συστήματος Διαχείρισης Βάσεων Δεδομένων (Database Management System-DBMS) με άλλες γλώσσες. Αξιοσημείωτες πλατφόρμες βάσεων δεδομένων είναι η Firebird (PSQL), Oracle (PL/SQL), MySQL (SQL/ Persistent Stored Module (PSM)) και Postgres (ή PostgreSQL) (PL/pgSQL) την οποία και επιλέξαμε να ασχοληθούμε.

### 3.4 PostgreSQL



#### 3.4.1 Γενικές Πληροφορίες

Η PostgreSQL είναι μια σχεσιακή βάση δεδομένων ανοικτού κώδικα. Αναπτύσσεται περισσότερο από δυο δεκαετίες και πλέον θεωρείται η πιο προηγμένη βάση δεδομένων ανοικτού κώδικα.

Υποστηρίζεται από πολλά λειτουργικά συστήματα όπως Windows, Linux και Unix. Παρέχει τη δυνατότητα δημιουργίας νέων τύπων δεδομένων ενώ παρέχει από μόνη της πάρα πολλά. Υποστηρίζει επίσης όλων των τύπων ενώσεων. Επιτρέπει στους χρήστες να γράφουν λειτουργίες σε μεγάλη ποικιλία γλωσσών, όπως Python, C, C++ κ.α. Παρέχει συγκεντρωτικές συναρτήσεις καθώς και τη δυνατότητα δημιουργίας νέων. Δημιουργία προσωρινών πινάκων που στη συνέχεια διαγράφονται όπως και απεριόριστο σχεδόν χώρο στους πίνακες αποθήκευσης δεδομένων.

### 3.4.2 Ιστορία

Η PostgreSQL (ή Postgres) ξεκίνησε το 1986 ως Postgres, ένα ερευνητικό πρόγραμμα του Πανεπιστημίου της Καλιφόρνια στο Μπέρκλεϋ, με επικεφαλής τον ερευνητή Michael Stonebraker. Στο σχεδιασμό του Postgres, η ομάδα του Stonebraker επεδίωξε να βελτιώσει το Ingres, ένα πρωτότυπο έργο υπό την ηγεσία του, κυρίως μέσω της υποστήριξης καθορισμένων από τον χρήστη τύπων (ή "domains") με αυθαίρετα σύνθετους επιχειρηματικούς κανόνες και άλλες έννοιες όσον αφορά σχέση-αντικείμενο.

Στόχος της Postgres ήταν να προσθέσει τα λιγότερα χαρακτηριστικά και δυνατότητες που απαιτούνταν για την πλήρη υποστήριξη τύπων δεδομένων. Αυτά τα χαρακτηριστικά περιλάμβαναν την ικανότητα ορισμού τύπων και πλήρη περιγραφή σχέσεων. Στην Postgres, εισήχθησαν διάφοροι κανόνες έτσι ώστε η βάση δεδομένων να κατανοεί τις σχέσεις και να ανακτούνται οι πληροφορίες σε συσχετισμένους πίνακες. Στην Postgres χρησιμοποιήθηκαν πολλές από τις ιδέες της Ingres, αλλά όχι ο κώδικας της αυτούσιος.

Η ομάδα του Stonebraker ανέπτυξε ενεργά την Postgres για οκτώ χρόνια και το έκανε χρησιμοποιώντας τη δική τους γλώσσα αναζήτησης, την Postquel. Ενώ θεωρούταν ανώτερη από την κυρίαρχη SQL, με μεγαλύτερο βάθος έκφρασης λόγω των πιο προηγμένων θεωρητικών της βάσεων, στην πράξη η Postquel δεν ευθυγραμμίστηκε με τις ανάγκες της βιομηχανίας, η οποία είχε ήδη τυποποιήσει την SQL. Για το λόγο αυτό, το 1995 δύο διδακτορικές σπουδές, οι μαθητές από το εργαστήριο του Stonebraker, Andrew Yu και Jolly Chen, αντικατέστησαν την Postquel με ένα εκτεταμένο υποσύνολο SQL. Η Postgres μετονομάστηκε στη συνέχεια σε Postgres95. Το 1996, το έργο μετονομάστηκε σε PostgreSQL για να αντικατοπτρίζει την υποστήριξή του στην SQL. Η PostgreSQL ξεκίνησε στην έκδοση 6, για λόγους συνέπειας με την έκδοσή του Berkeley. Από τότε κι έπειτα μια ομάδα προγραμματιστών βάσεων δεδομένων που εργάζονται κυρίως στον κλάδο καθώς και εθελοντών ανά τον κόσμο ανέλαβε τον έλεγχο, τη συντήρηση και τη συνεχή εξέλιξη του κώδικα Postgres.

### 3.4.3 Εγκατάσταση PostgreSQL

Για την εγκατάσταση της PostgreSQL χρειάστηκε πρώτα να κατεβάσουμε το αρχείο από την ιστοσελίδα <https://www.enterprisedb.com/downloads/postgres-postgresql-downloads#windows>. Στη συνέχεια ακολουθώντας τις οδηγίες από το wizard που προσφέρει έγινε η εγκατάσταση στο σύστημα. Δεν χρειάστηκε να εγκαταστήσουμε κάποιο επιπλέον πακέτο από αυτά που μας προσφέρει οπότε προχωρήσαμε με την απλή εγκατάσταση της. Πολύ σημαντικό είναι ότι κατά την διάρκεια της εγκατάστασης το πρόγραμμα μας ζητάει έναν κωδικό, καλό θα ήταν αυτός ο κωδικός να είναι “ισχυρός” αλλά και κάποιος που δεν θα ξεχάσουμε στο μέλλον γιατί χωρίς αυτόν δεν θα μπορούμε να έχουμε πρόσβαση στο πρόγραμμα και στην βάση δεδομένων που πρέπει να δημιουργήσουμε.. Αφού ολοκληρώθηκε η εγκατάσταση, συνέχεια πήρε η δημιουργία της βάσης όπου χρησιμοποιήθηκε για την ιστοσελίδα. Αφού ανοίξουμε το pgAdmin που είναι το πρόγραμμα διαχείρισης της βάσης που προσφέρει η PostgreSQL, δίνουμε τον κωδικό που έχουμε ορίσει για το χρήστη των βάσεων που έχει. Στη συνέχεια στο πεδίο databases κάνουμε δεξί κλικ και δημιουργία νέας βάσης, εκεί δίνουμε τα στοιχεία που επιθυμούμε και πατάμε αποθήκευση.

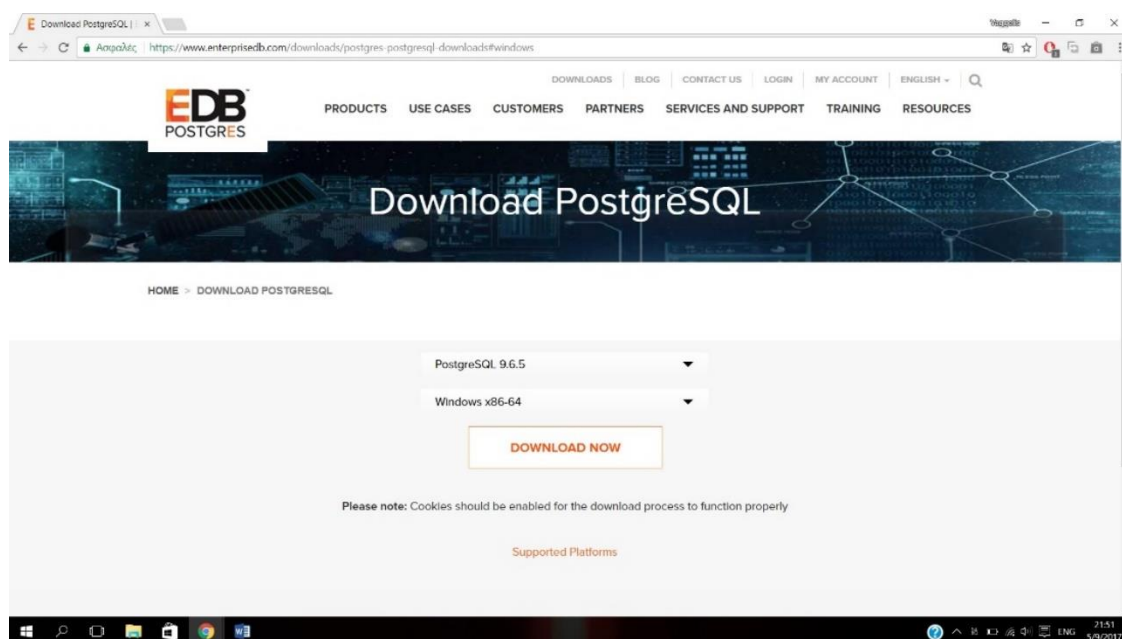
Παρακάτω βλέπουμε την ιστοσελίδα της PostgreSQL <https://www.postgresql.org/>

The screenshot shows the PostgreSQL website's 'Windows installers' page. The page title is 'Windows installers' and it features a navigation menu with links for Home, About, Download, Documentation, Community, Developers, Support, and Your account. The main content area is titled 'Interactive installer by EnterpriseDB' and provides information about the installer, including its features and platform support. A table lists supported PostgreSQL versions and their corresponding 64-bit and 32-bit Windows platforms.

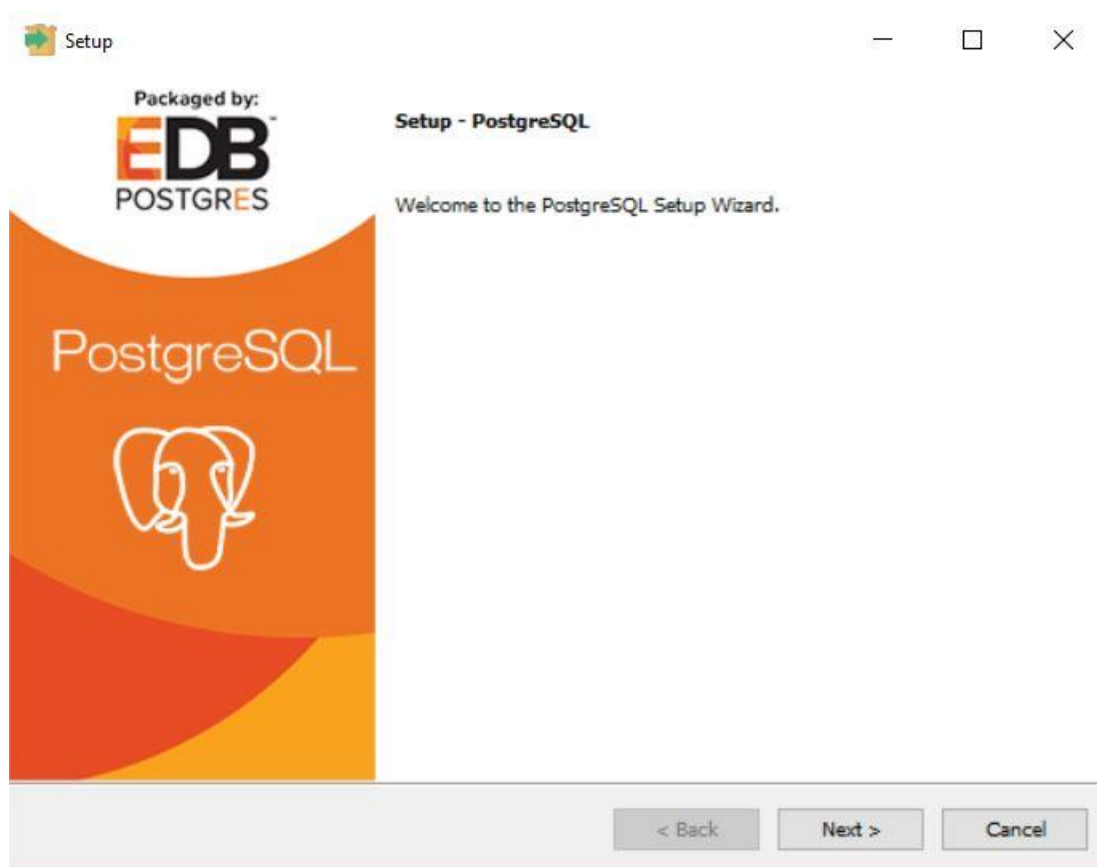
PostgreSQL Version	64 Bit Windows Platforms	32 Bit Windows Platforms
9.6	2012 R2 & R1, 2008 R2, 7, 8, 10	2008 R1, 7, 8, 10
9.5	2012 R2 & R1, 2008 R2	2008 R1
9.4	2012 R2, 2008 R2	2008 R1
9.3	2012, 2008 R2	2008 R1
9.2	2008 R2 & R1	2008 R1

Below the table, there is a section for 'Graphical installer by BigSQL' which provides a link to download the graphical installer and a brief description of its contents.

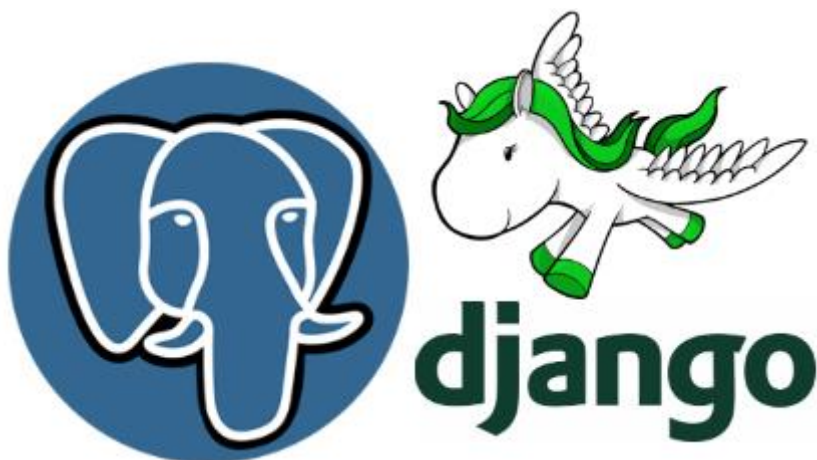
Η ιστοσελίδα <https://www.enterprisedb.com/downloads/postgres-postgresql-downloads#windows> από όπου θα κατεβάσουμε την εφαρμογή pgAdmin 4 για την δημιουργία και διαχείριση της βάσης δεδομένων μας..



Παρακάτω βλέπουμε την εγκατάσταση της εφαρμογής..



## 3.5 Django &amp; PostgreSQL



Για τη σύνδεση της PostgreSQL στο Django χρειάστηκε ο adapter `psycopg2`. Μετά από αυτό πηγαίνοντας στο αρχείο `setting.py`. Στο σημείο όπου εκχωρούνται τα στοιχεία της βάσης που επιθυμούμε να συνδέσουμε, τα διαμορφώνουμε ως εξής :

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql',
        'NAME': 'testdatabase1',
        'USER': 'postgres',
        'PASSWORD': 'exampleCode',
        'HOST': 'localhost',
        'PORT': '5433',
    }
}
```

Η PostgreSQL “ακούει” στις πόρτες 5432 ή 5433 οπότε πρέπει να υπάρξει προσοχή για να δώσουμε τη σωστή πόρτα.

Στη συνέχεια από το `cmd` γράφουμε το `path` στον φάκελο που βρίσκετε το `project` για να ολοκληρώσουμε τη σύνδεση με τη βάση. Αυτό γίνεται με τη χρήση των παρακάτω εντολών με τη σειρά που είναι.

#### 1: `python manage.py makemigrations`

Αυτή η εντολή δημιουργεί τις αλλαγές ή δημιουργία νέων μοντέλων για την βάση και τα τροποποιεί σε μια μορφή έτοιμη για να αποθηκευτούν στη βάση.

#### 2: `python manage.py migrate`

Αυτή η εντολή είναι αυτή όπου αποθηκεύει τελικά στη βάση όποιες αλλαγές έχουν γίνει από την προηγούμενη εντολή.



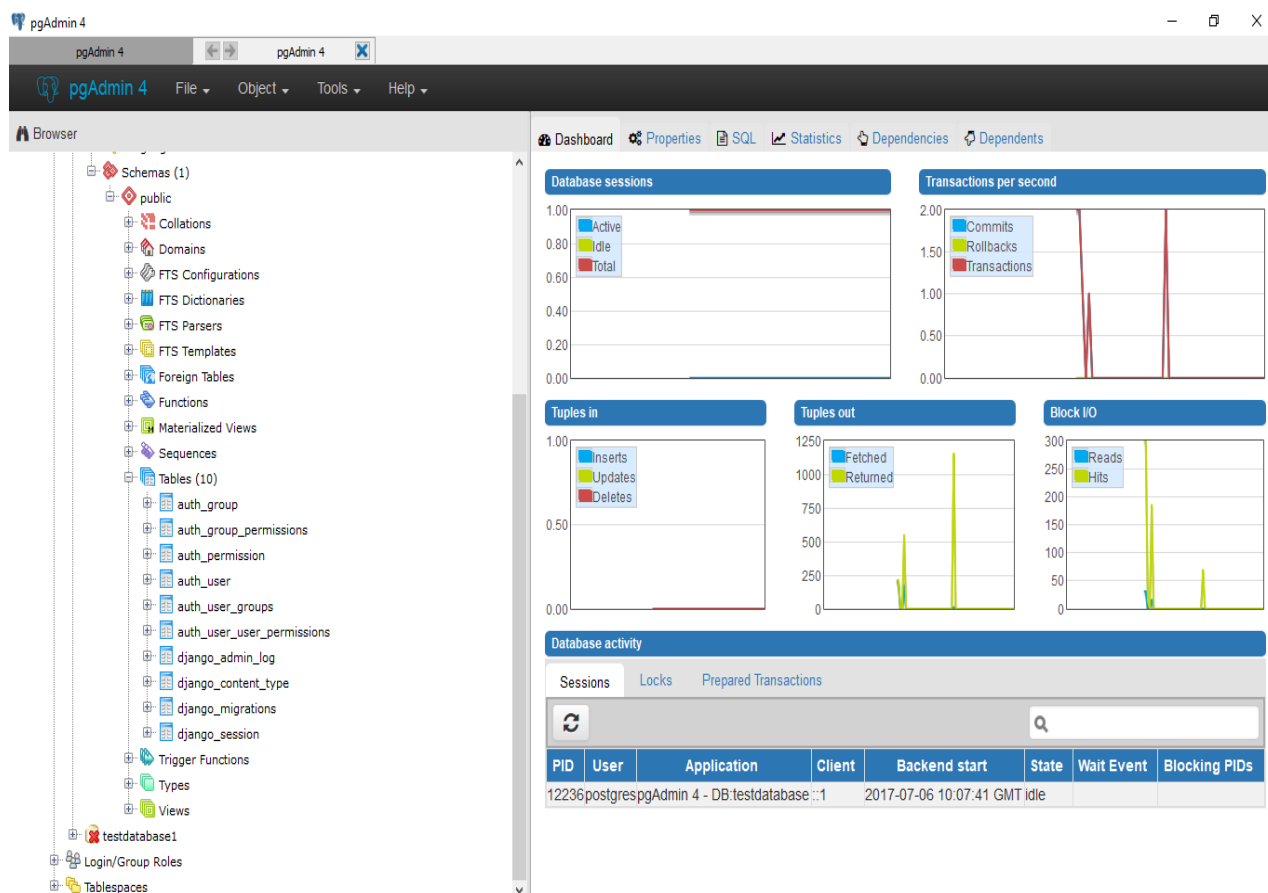
Την πρώτη φορά που θα επιτευχθεί η σύνδεση θα ζητηθεί η δημιουργία ενός superuser εάν δεν υπάρχει.

Η εντολή για αυτό είναι : **python manage.py createsuperuser**.

Συμπληρώνοντας τα στοιχεία που ζητάει δημιουργείτε ο superuser όπου με τα στοιχεία αυτά μπορεί να πραγματοποιηθεί login στη συνέχεια στην ιστοσελίδα ως admin.

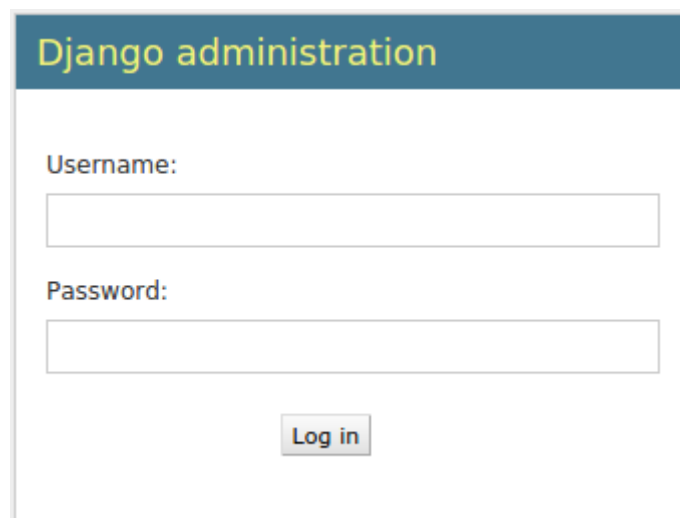
Όταν ολοκληρωθεί η διαδικασία της σύνδεσης θα πρέπει στη βάση δεδομένων να έχουν δημιουργηθεί κάποια tables που εισάγει το Django από μόνο του για την διευκόλυνση του χρήστη.

Τα αρχικά tables που δημιουργεί το Django φαίνονται στην παρακάτω εικόνα:



Σε αυτό το σημείο εάν τρέξει το project από το visual studio ή με την εντολή `python manage.py runserver 0.0.0.0:8000` όπου τα 0.0.0.0 αντιπροσωπεύουν το domain ή ip της ιστοσελίδας και το 8000 την προκαθορισμένη σελίδα του Django, θα πρέπει να μας εμφανίσει την αρχική σελίδα του Django.

Στη συνέχεια άμα προστεθεί σε αυτό το url το /admin θα πρέπει να εμφανίσει μια σελίδα για log in όπως φαίνεται παρακάτω:



The image shows a screenshot of the Django administration login page. At the top, there is a dark blue header with the text "Django administration" in yellow. Below the header, there are two input fields: "Username:" and "Password:". Below the password field, there is a "Log in" button.

## 4. Κύριο μέρος Πτυχιακής

### 4.1 Ανάλυση προβλήματος

Ο λόγος της δημιουργίας της ιστοσελίδας είναι η εύκολη ενημέρωση καθώς και η εύκολη κράτηση ξενοδοχείων και υπηρεσιών για όσους επιθυμούν να επισκεφτούν την Κρήτη. Μέσω της ιστοσελίδας αυτής μπορούν να ενημερωθούν για αξιοθέατα και μουσεία καθώς και για άλλες δραστηριότητες κι εκδρομές που λαμβάνουν μέρος σε περιοχές της Κρήτης. Τους παρέχεται η δυνατότητα να κάνουν κράτηση σε ξενοδοχεία, αεροπορικά εισιτήρια ή караβιού όπως και η κράτηση μεταφορικού μέσου. Στην ιστοσελίδα υπάρχουν προσφορές με πακέτα σε πιο οικονομική τιμή.

### 4.2 Δημιουργία App

Για τη δημιουργία ενός app κάνουμε δεξί κλικ στο Django project και στη συνέχεια add και Django app. Αφού δώσουμε όνομα πατάμε enter και αυτό δημιουργεί το app για εμάς, μαζί με μερικούς φακέλους που θα χρειαστούμε.

Στο φάκελο settings.py του Django πρέπει να δηλωθεί το project που θα δημιουργηθεί ως app, στο αντίστοιχο σημείο ως :

```
INSTALLED_APPS = [
    # Add your apps here to enable them
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'services',
]
```

Όπου services είναι το app.

Στη συνέχεια πρέπει να δηλωθεί και το path που θα περιέχει τα templates:

```
TEMPLATE_DIRS = (
    'ptuxiakh\DjangoWebProject1\DjangoWebProject1\services\templates',
)
```

)

Τέλος στο φάκελο urls.py θα μπουν τα urls του app όπου θα ξεκινάει και το app.

```
urlpatterns = [
    url(r'^$', lambda r: HttpResponseRedirect('services/home')),
    url(r'^services/', include('services.urls')),
    url(r'^admin/', include(admin.site.urls)),
]
```

### 4.3 Δημιουργία μοντέλων

Η δημιουργία των μοντέλων για την βάση δεδομένων στο Django είναι δυνατό να δημιουργηθούν όπως απλές classes στον αντικειμενοστραφή προγραμματισμό, κάνοντας την δημιουργία τους πολύ εύκολη και κατανοητή. Παρόλο που η δημιουργία των μοντέλων είναι απλή αυτό δεν σημαίνει ότι θα μπορούσαμε να δηλώσουμε όλα τα στοιχεία που θέλουμε σε μια κλάση. Έτσι παραπάνω κλάσεις δημιουργήθηκαν για την ολοκλήρωση ενός μοντέλου.

Ένα παράδειγμα φαίνεται στον παρακάτω ενδεικτικό κώδικα όπου δείχνει πως δημιουργήθηκε το μοντέλο για τα ξενοδοχεία:

```
class Hotel(models.Model):
    name = models.CharField(max_length=200)
    adress = models.CharField(max_length=500)
    category = models.IntegerField(default = 1, choices = HOT_CAT)
    room_num = models.IntegerField(default=0)
    phone_num = models.CharField(max_length = 15, default=0)

    def __str__(self):
        return self.name

class HotelRooms(models.Model):
    hotel_name = models.ForeignKey(Hotel, on_delete=models.CASCADE)
    code = models.CharField(max_length = 7, default="hot", editable=False)
    room_category = models.IntegerField(default=1, choices = ROOM_CAT)
    room_cat_num = models.IntegerField(default=0)
    price = models.FloatField(default=0)

    def __str__(self):
        return '%s %s' % (self.hotel_name.name, self.room_category)
```

Στο category η μεταβλητή HOT\_CAT στην κλάση Hotel δηλώνει μία enum μεταβλητή για την επιλογή της κατηγορίας του ξενοδοχείου καθώς η συνάρτηση `__str__` έχει οριστεί για να επιτρέψει την εμφάνιση του ονόματος των ξενοδοχείων στη σελίδα του admin στον πίνακα με τα ξενοδοχεία.

Στη κλάση HotelRooms το πεδίο `hotel_name` είναι ξένο κλειδί από την κλάση Hotel. Έχει γίνει έτσι για να μπορούμε να προσθέτουμε σε κάθε ξενοδοχείο τις κατηγορίες δωματίων που έχει. Επίσης τον αριθμό δωματίων της κάθε κατηγορίας όπως και την τιμή τους. Το πεδίο `code` είναι ένα αναγνωριστικό string για να μπορεί να αναγνωριστεί ότι είναι τύπος ξενοδοχείου από ένα και μόνο πεδίο. Με αυτόν τον τρόπο γίνεται ευκολότερη η αναζήτηση στη βάση δεδομένων όταν ψάχνουμε για ξενοδοχεία ανάμεσα σε άλλους τύπους υπηρεσιών, όπως μεταφορικά μέσα ή εκδρομές.

Δεν είναι αρκετό όμως αυτό για να εμφανιστεί στη σελίδα του admin ο πίνακας με τα ξενοδοχεία. Πρέπει να προστεθεί στο φάκελο με όνομα `admin.py`.

Σε αυτόν τον φάκελο γίνεται η επιλογή ποιων πινάκων από την βάση θα παρουσιάζονται στη σελίδα του admin και πως. Με τις παρακάτω γραμμές κώδικα γίνεται η επιλογή να παρουσιάζετε το όνομα του ξενοδοχείου και η προσθήκη του στη σελίδα του admin στη συνέχεια:

```
class HotelAd(admin.ModelAdmin):
    list_display = ('name',)

admin.site.register(Hotel, HotelAd)
```

Το Hotel αντιπροσωπεύει το μοντέλο από το φάκελο `models.py` και σηματοδοτεί από που θα πάρει τις πληροφορίες για να παρουσιάσει στη σελίδα.

Στην παρακάτω εικόνα φαίνεται πως παρουσιάζεται στο admin page το παραπάνω:



Σε περίπτωση που περισσότερες πληροφορίες είναι αναγκαίο να εμφανιστούν ένα παράδειγμα κώδικα ακολουθεί με τον τρόπο που μπορεί να γίνει:

```
class combOf(admin.ModelAdmin):
    list_display = ('hotName', 'rentName', 'travelName')

    def hotName(self, obj):
        name = obj.hotel_room.hotel_name.name
        return name
    def rentName(self, obj):
        name = obj.travel.travelcomp.transpComp
        return name
    def travelName(self, obj):
        name = obj.rentCars.rentOffice.name
        return name
```

Ακολουθεί εικόνα που δείχνει πως παρουσιάζεται στο admin page το παραπάνω:

The screenshot shows the Django administration interface for 'Combo offers'. At the top, it says 'Django administration' and 'WELCOME, KORSIK. VIEW SITE / CHANGE PASSWORD / LOG OUT'. Below that, the breadcrumb is 'Home > Services > Combo offers'. The main heading is 'Select combo offers to change' with an 'ADD COMBO OFFERS +' button. There is an 'Action:' dropdown menu and a 'Go' button, with '0 of 1 selected' next to it. A table with one row is shown below:

HOTNAME	RENTNAME	TRAVELNAME
<input type="checkbox"/> minoan palace	caravel	aegean

Below the table, it says '1 combo offers'.

Τελειώνοντας έτσι με την εισαγωγή πινάκων στη βάση και με την εμφάνιση τους στη σελίδα του admin για τη δυνατότητα προσθήκης ή αλλαγής των εγγραφών, ακολουθεί η δημιουργία των σελίδων όπου θα έχει πρόσβαση ο χρήστης.

#### 4.4 Templates

Στο project σε ένα φάκελο κρατάμε όλα τα templates που θα χρειαστούν (services/templates). Ξεκινώντας από ένα γενικό template όπου πάνω σε αυτό θα βασιστούν όλα τα υπόλοιπα, το layout1. Εκεί υπάρχουν τα headers, footers και το κύριο μέρος της σελίδας.

##### 4.4.1 Template bracket

Το Django προσφέρει τη δυνατότητα με τη χρήση ειδικών “brackets” τον ορισμό και διαχωρισμό των headers, footers κτλ. Έτσι είναι δυνατή η τροποποίηση τους και η προσθήκη κώδικα σε συγκεκριμένα σημεία μόνο με τη χρήση αυτών. Στο layout1 υπάρχουν αυτά τα “brackets” με το ένα “bracket” να είναι το main content όπου είναι κενό. Στη δημιουργία των υπόλοιπων templates καλείτε το layout1 με τρόπο που θυμίζει πολύ την κληρονομικότητα σε αντικειμενοστραφή προγραμματισμό, και απλά με τη χρήση των “brackets” συμπληρώνεται κώδικας όπου χρειάζεται.

Ένα παράδειγμα αυτών των “brackets” : `{% block content %} {% endblock %}` .

Με τον ίδιο τρόπο γράφουμε τα if :

```
{% if %}

{% else %}

{% endif %}

{% endblock %}
```

Αρχίζοντας τη δημιουργία των υπόλοιπων templates δηλώνουμε στην αρχή το βασικό template layout1 και τον τίτλο της σελίδας με τον παρακάτω τρόπο.

```
{% extends 'layout1.html' %}

{% block title %} Renting page {% endblock %}
```

Στη συνέχεια μέσα στο content block γράφουμε ότι χρειαζόμαστε για το κάθε template. Όπως πίνακες, κουμπιά, πεδία κειμένου καθώς και το φόρτωμα και την αποστολή τιμών από τη βάση, όπου βοηθάνε στη χρήση του site.



#### 4.4.2 Security token

Για την αποστολή τιμών όμως το Django μας αναγκάζει να χρησιμοποιήσουμε ένα token που μας παρέχει έτσι ώστε να διασφαλίσει ότι τα δεδομένα μας θα είναι ασφαλή. Το token αυτό το δηλώνουμε όπως και τα Blocks και παρουσιάζεται παρακάτω.

```
{% csrf_token %}
```

Τα αρχικά του σημαίνουν Cross Site Request Forgeries όπου είναι ένας τύπος επίθεσης που μέσω κάποιας φόρας ή κουμπιού μπορεί να προβεί σε ανεπιθύμητες ενέργειες με τα στοιχεία του χρήστη. Το csrf\_token το χρησιμοποιούμε μέσα στη φόρμα που θέλουμε να προστατέψουμε με τον τρόπο που δείχνουμε παραπάνω.

#### 4.4.3 Html, css & images

Εφόσον προσθέσουμε και την προστασία που απαιτεί το Django συνεχίζουμε τη δημιουργία των templates με απλό html κώδικα. Για να κάνουμε πιο εμφανίσιμα τα template χρησιμοποιούμε τη βοήθεια του css ( Cascading StyleSheets). Με το css μπορούμε να δώσουμε χρώμα, σχήμα, διάφορα εφέ όπως θολούρα και σκιές, μπορούμε επίσης να κάνουμε να έχουν διαφορετική συμπεριφορά τα κουμπιά ανάλογα με το που βρίσκεται το ποντίκι και πολλά άλλα. Το css μας βρίσκεται σε δύο αρχεία κάτω από τους φακέλους static/services. Το tableDes1 όπου έχουμε το design για τα tables/ πίνακες και το αρχείο stylingTest. Στο stylingTest βρίσκονται βασικές πληροφορίες για τα templates καθώς και εικόνες που έχουν χρησιμοποιηθεί όπως για τα κουμπιά. Οι εικόνες που βρίσκονται στις σελίδες που στόχος τους είναι να προσδιορίζουν το περιεχόμενο της κάθε σελίδας, είναι ορισμένες στο ίδιο αρχείο css. Τέλος περιέχει τα navigation bars που υπάρχουν στα templates και τα drop down menu.

Ένα παράδειγμα από ένα “block” css :

```
.hotIm{
  background-image: url("../images/hotel.jpg");
  background-size: 100% 200px;
  background-repeat: no-repeat;
  padding: 100px;
}
```

Το παραπάνω μας περιγράφει τον τρόπο που θα γίνεται η εισαγωγή μιας εικόνας στο template.

Στο φάκελο static έχουμε επίσης άλλο ένα φάκελο στον οποίο έχουμε αποθηκεύσει όλες τις εικόνες που χρησιμοποιούνται για την ιστοσελίδα.

#### 4.4.4 Template URLs

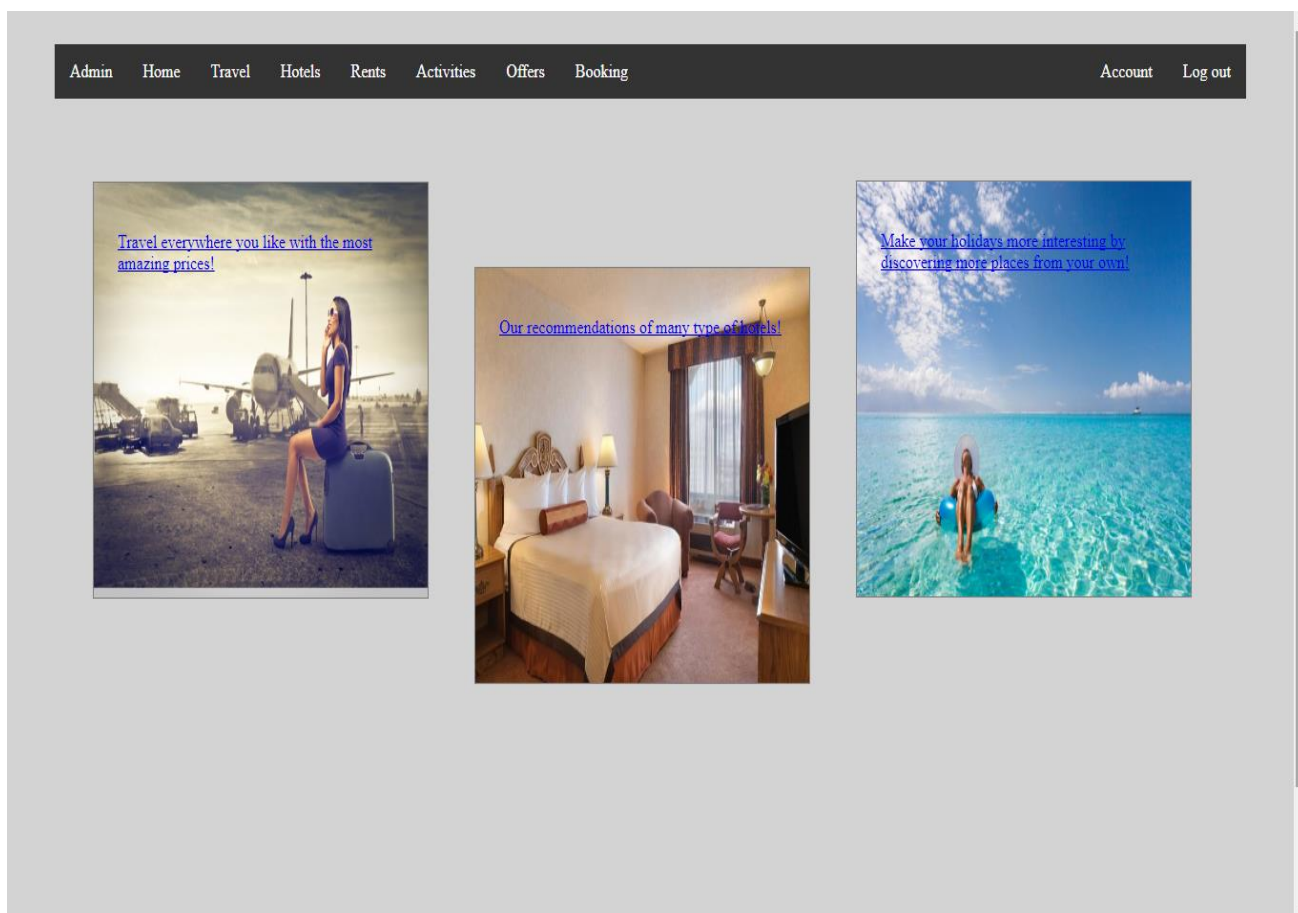
Για κάθε template που δημιουργήσαμε πρέπει να το δηλώσουμε στα urls του Project μας ώστε να δημιουργήσουμε τον σύνδεσμο για να μπορεί να γίνει redirect στη σελίδα που ζητάμε από το menu ή από άλλα μέσα του project. Η λίστα με τα urls φαίνεται παρακάτω και περιέχει όλα τα urls που χρησιμοποιήθηκαν.

```
urlpatterns = [  
    url(r'^home', views.home, name='services'),  
    url(r'^hotel', views.hotel, name='hotel'),  
    url(r'^rent', views.rent, name='car'),  
    url(r'^museum', views.museum, name='museum'),  
    url(r'^sight', views.sight, name='sight'),  
    url(r'^excursion', views.excursion, name='trip'),  
    url(r'^travel', views.travel, name='travel'),  
    url(r'^offers', views.offers, name='offers'),  
    url(r'^booking', views.booking, name='booking'),  
    url(r'^myAccount', views.myAccount, name='myAccount'),  
    url(r'^login', auth_views.login, {'template_name': 'login.html'}, name = 'login'),  
    url(r'^logout', auth_views.logout, {'template_name': 'logout.html'}, name = 'logout')  
]
```

#### 4.4.5 Template Images

Στη συνέχεια θα δείξουμε εικόνες από τα template που υπάρχουν κατά την πορεία του project. Ένας σύντομος σχολιασμός θα υπάρχει για κάθε ένα από αυτά για να περιγράψει τι φαίνεται στην εικόνα καθώς και τους λόγους που έγιναν έτσι.

Παρακάτω φαίνεται το homepage όπου υπάρχει σε αυτό ένα Link στις εικόνες που παραπέμπει τον χρήστη σε αντίστοιχες σελίδες του site.



Το travel page όπου υπάρχουν όλες οι διαδρομές και οι εταιρίες που έχει το site. Μπορεί να γίνει αναζήτηση στο πλάι βάση διάφορων επιλογών όπως ημερομηνία, εταιρία, μέσω μεταφοράς και μέρος προορισμού ή αναχώρησης. Παρουσιάζονται πολύ απλά και ξεκάθαρα έτσι ώστε ο χρήστης να μην χρειάζεται να ψάχνει που θα βρει το κάθε τι.

The screenshot shows a travel website interface. At the top, there is a navigation menu with links: Admin, Home, Travel, Hotels, Rents, Activities, Offers, Booking, Account, and Log out. Below the menu is a world map with several red location markers. To the left of the map is a search and select form with fields for Mean of transport, Travel company, Destination, and Departure date, along with a Submit button. To the right of the form is a table with the following data:

	Travel Company	From Start Point	To Destination	Departure	Arrival	Price
<input checked="" type="checkbox"/>	aegean	heraklion	athens	Feb. 21, 2017 1:31 p.m.	Feb. 21, 2017 2:31 p.m.	75.0
<input checked="" type="checkbox"/>	aegean	athens	heraklion	Feb. 21, 2017 10:34 p.m.	Feb. 21, 2017 11:34 p.m.	75.0


Ενδεικτικά έχουν τοποθετηθεί κάποιες καταχωρίσεις στη βάση ώστε να φαίνεται το αποτέλεσμα.

Σε αυτό το template, που είναι το hotel page έχουμε τα δωμάτια που προσφέρουν τα ξενοδοχεία που υπάρχουν στο site. Όπως και με το Travel page έτσι κι εδώ το Template είναι παρόμοιο, προσφέροντας επιλογές αναζήτησης βάσει κάποιων κριτηρίων. Σε αυτό το site όμως για την αναζήτηση δωματίου ο χρήστης καλείται να δώσει την ημερομηνία που επιθυμεί να κλίσει κάποιο δωμάτιο έτσι ώστε να του παρουσιαστούν μόνο όσα είναι διαθέσιμα στις ημερομηνίες που θέλει.

	Hotel	Adress	Category	Room type	Phone number	Price
<input checked="" type="checkbox"/>	minoan palace	platanias	5 Stars	Suites	2821069315	3000.0/day
<input checked="" type="checkbox"/>	platanias beach	platanias	3 Stars	2 Beds	7635457	80.0/day

Παραπάνω φαίνεται η σελίδα πριν την επιλογή ημερομηνιών από τον χρήση.

Admin Home Travel Hotels Rents Activities Offers Booking
Account Log out



Search & Select

08/14/2017  
Arrival date

08/17/2017  
Departure date

**Hotel category:**

3 Stars  
 4 Stars  
 5 Stars


**Room types**

	Hotel	Adress	Category	Room type	Phone number	Price
<input checked="" type="checkbox"/>	minoan palace	platanias	5 Stars	Suites	2821069315	3000.0/day
<input checked="" type="checkbox"/>	platanias beach	platanias	3 Stars	2 Beds	7635457	80.0/day

Παραπάνω φαίνεται η σελίδα μετά την επιλογή ημερομηνιών από τον χρήστη.

Παρακάτω βρίσκετε το rent page. Εδώ ο χρήστης μπορεί να ελέγξει και να επιλέξει μεταφορικό μέσο για ενοικίαση. Το menu είναι παρόμοιο με τα προηγούμενα καθώς κι εδώ ο χρήστης πρέπει να επιλέξει ημερομηνίες που επιθυμεί να ενοικιάσει. Μετά προαιρετικά προσφέρονται περισσότερες επιλογές αναζήτησης όπως ο τύπος του μεταφορικού μέσου.

Admin
Home
Travel
Hotels
Rents
Activities
Offers
Booking
Account
Log out



Search & Select

08/14/2017  
Arrival date

08/17/2017  
Departure date

Rent a bike  
 Rent a car

**Car categories**

City small  
 City large  
 Jeep  
 Wagon  
 Lux

**Bike categories**

Scooter  
 Bikes

	Vehicle Model	Cubic	Passenger Number	Category	Rent Office	Price
<input checked="" type="checkbox"/>	getz	1400	4	City small	caravel	from 25.0/day
<input checked="" type="checkbox"/>	i10	1000	4	Jeep	caravel	from 30.0/day
<input checked="" type="checkbox"/>	3'	1600	5	City large	caravel	from 35.0/day

Στο excursion page όπου βρίσκετε σαν υποκατηγορία στο activities page στο κεντρικό menu μπορούμε να βρούμε διάφορες εκδρομές που είναι διαθέσιμες στην ιστοσελίδα όπως επίσης πακέτα προσφορών με πολλές εκδρομές μαζί σε καλύτερη τιμή.

Search & Select

Search Destination!

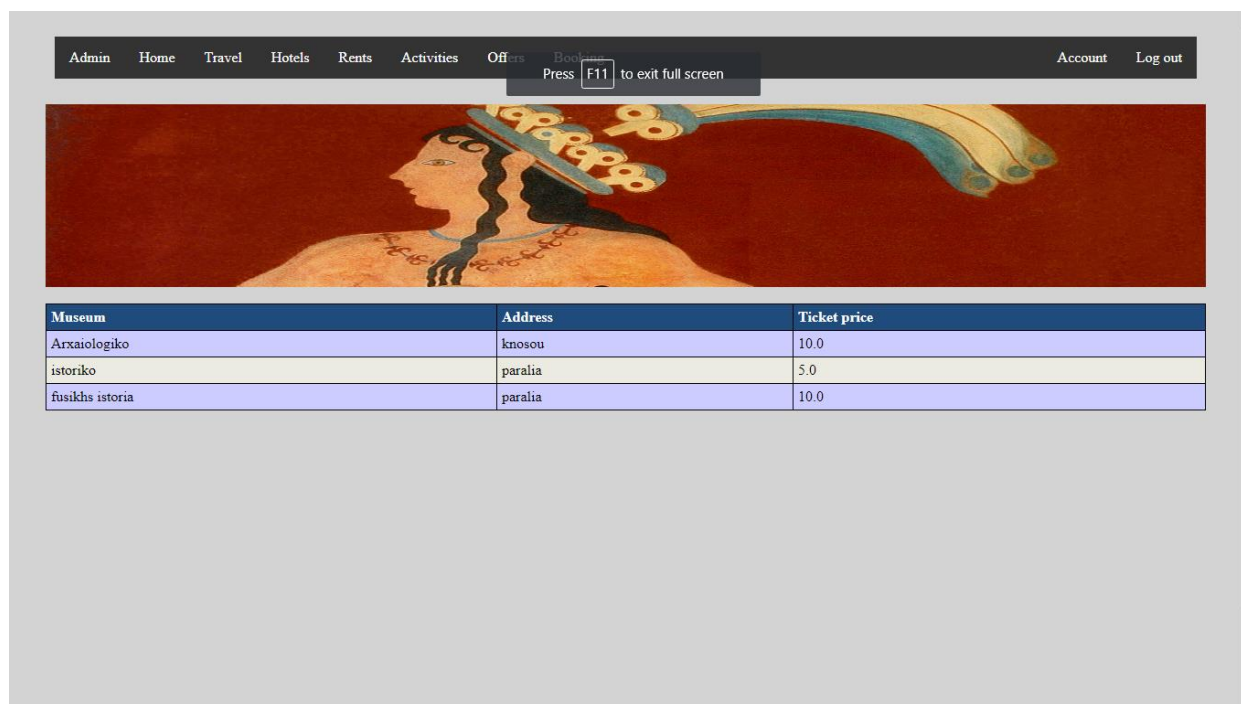
Submit

	Destination	Start point	Departure	Return	Office name	Price
<input checked="" type="checkbox"/>	bottom of the ocean	kalathas	Feb. 23, 2017 11:30 a.m.	Feb. 23, 2017 12:30 p.m.	adveturing	200.0

	Office Name	Offer type	Price
<input checked="" type="checkbox"/>	adveturing	3 Excursions	300.0



Οι επόμενες δύο υποκατηγορίες στο activities page είναι οι museums και sights. Σε αυτές υπάρχουν πληροφορίες για τα μουσεία και τα αξιοθέατα που βρίσκονται στο νησί. Λίγες πληροφορίες δίνονται για αυτά όπως το που βρίσκονται καθώς και η τιμή εισιτηρίου εάν υπάρχει. Δεν υπάρχει η επιλογή κράτησης εισιτηρίων για αυτά μιας και για τα αξιοθέατα δεν υπάρχει καν εισιτήριο και στα μουσεία δίνονται εισιτήρια στην είσοδο τη στιγμή που πάει ο επισκέπτης.




The screenshot shows a website interface with a dark navigation bar at the top containing links: Admin, Home, Travel, Hotels, Rents, Activities, Offices, Bookings, Account, and Log out. A tooltip over the Bookings link says 'Press [F11] to exit full screen'. Below the navigation is a large banner image of a woman in traditional attire. Underneath the banner is a table with the following data:

Museum	Address	Ticket price
Arxaiologiko	knosou	10.0
istoriko	paralia	5.0
fusikhs istoria	paralia	10.0

Παραπάνω φαίνεται η σελίδα με τα μουσεία.

Admin Home Travel Hotels Rents Activities Offices Bookings
Account Log out

Press F11 to exit full screen




Sight	Address	Ticket price
liontaria	kentro, hrakleio	0.0
frourio koules, hrakleio	paralia	0.0

Σελίδα με ενδεικτικά αξιοθέατα στο νησί.

Η παρακάτω σελίδα, το offers page, παρουσιάζει διάφορες προσφορές – πακέτα που μπορεί να προσφέρει το site σε συνδυασμό με τις εταιρίες που συνεργάζεται. Έτσι φαίνονται παρακάτω μια ενδεικτική προσφορά με ξενοδοχείο, εννοκίαση αμαξιού και μεταφορά στον προορισμό. Η προσφορές ισχύουν για συγκεκριμένες ημερομηνίες και σε συγκεκριμένη διαθεσιμότητα.

Admin Home Travel Hotels Rents Activities Offers Bookings
Account Log out

Press **F11** to exit full screen



Search & Select

Search Destination!

Submit

Hotel	Rent	Transport Company	Departure	Destination	Dates	Prices
<input checked="" type="checkbox"/> minoan palace, 5 Stars, Suites	caravel, getz	aegean	heraklion	athens	Feb. 21, 2017 to Feb. 25, 2017	3200.0

Στο booking page φαίνονται όσα έχει επιλέξει ο χρήστης για κράτηση και το συνολικό ποσό που θα πληρώσει για αυτά σύμφωνα με τις μέρες που έχει κάνει κράτηση. Έπειτα μόλις πατήσει book η κράτηση του καταχωρείτε στη βάση και ανάλογες ενέργειες γίνονται στα προϊόντα του site όπως η αφαίρεση διαθεσιμότητας του προϊόντος.

The screenshot shows a web interface for a booking system. At the top, there is a navigation menu with links: Admin, Home, Travel, Hotels, Rents, Activities, Offers, Bookings, Account, and Log out. A tooltip over the Bookings link says "Press F11 to exit full screen". Below the menu is a large image of a woman in traditional attire. The main content area is titled "Bookings" and contains a table with the following data:

	Description	price
<input checked="" type="checkbox"/>	caravel, getz	25.0/day
<input checked="" type="checkbox"/>	minoan palace, Suites	3000.0

On the left side, there is a sidebar with the following elements:

- Bookings
- 6075.0
- Total Price
- Book
- Book Now!

Στο account page φαίνονται πληροφορίες που αφορούν τις κρατήσεις του χρήστη. Από εδώ ο χρήστης μπορεί να ακυρώσει κάποια κράτηση που έχει κάνει και δεν επιθυμεί να την έχει πλέον.

The screenshot shows a user account page with a dark navigation bar at the top. The navigation bar contains links for Admin, Home, Travel, Hotels, Rents, Activities, Offers, Bookings, Account, and Log out. A tooltip is visible over the Bookings link, stating 'Press F11 to exit full screen'. Below the navigation bar, the user is greeted with 'Welcome korsik!'. The main content area is titled 'Your Bookings' and contains a table with the following data:

	Book	Departue date	Arrival date	Price
<input checked="" type="checkbox"/>	car-2	March 14, 2017	March 22, 2017	200.0
<input checked="" type="checkbox"/>	car-1	Aug. 14, 2017	Aug. 17, 2017	75.0
<input checked="" type="checkbox"/>	hot-1	Aug. 14, 2017	Aug. 16, 2017	6000.0

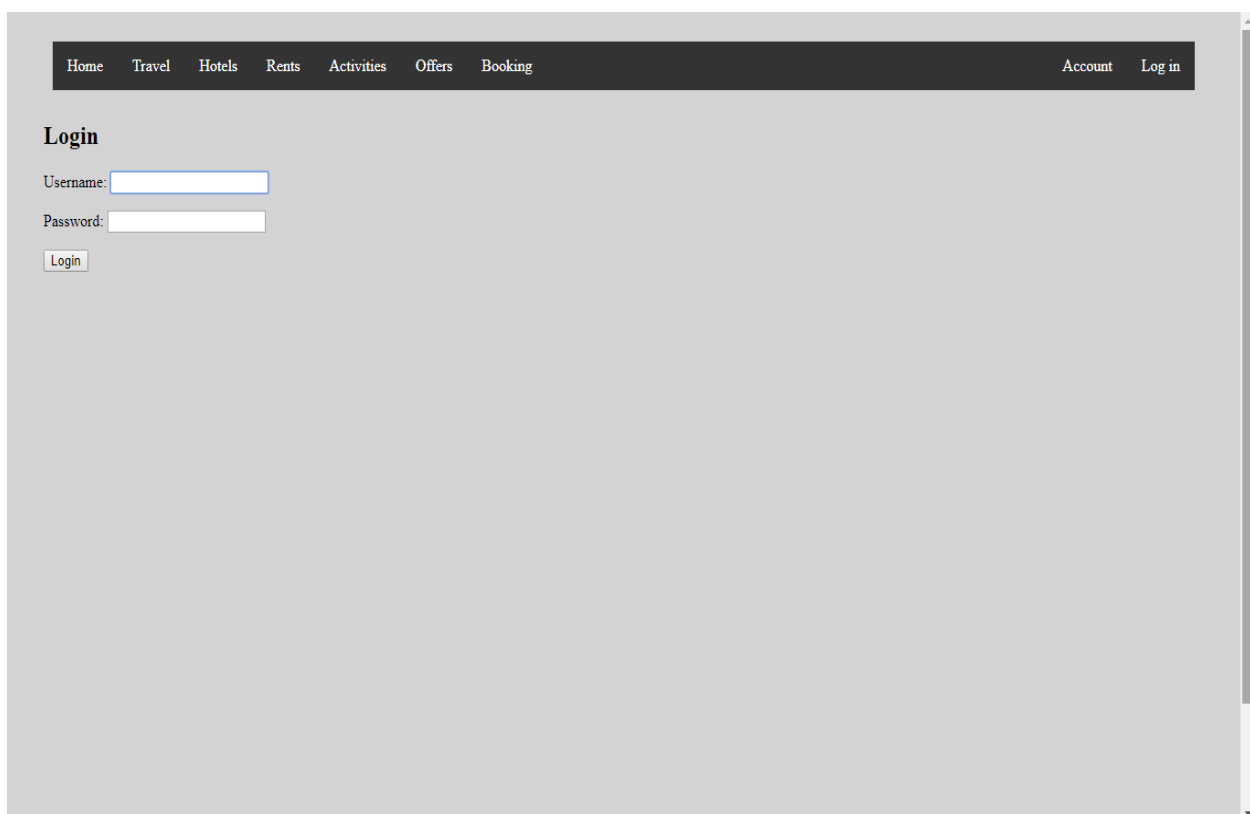
Στο login page χρησιμοποιούμε την φόρμα που μας προσφέρει έτοιμη το Django. Αυτό το κάνουμε στα urls του Project μας όπου χρησιμοποιώντας την βιβλιοθήκη του Django `django.contrib.auth` κάνουμε Import τα views. Στα urls το δηλώνουμε ως :

```
url(r'^login', auth_views.login, {'template_name': 'login.html'}, name = 'login'),
```

Όπου `auth_views` είναι τα views από τη βιβλιοθήκη.

Με τον ίδιο τρόπο καλούμε και το logout page.

Παρακάτω φαίνεται εικόνα με το login page



Home Travel Hotels Rents Activities Offers Booking Account Log in

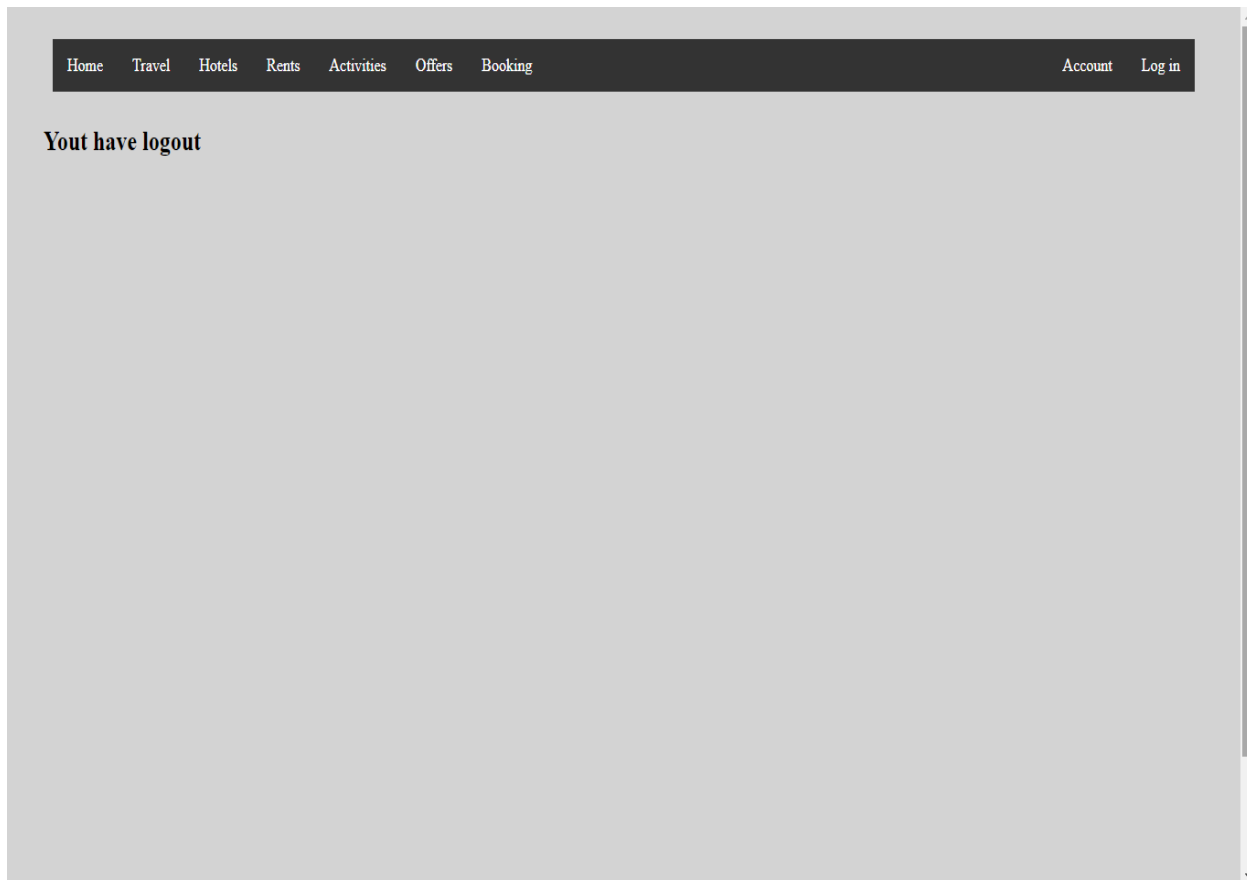
**Login**

Username:

Password:

Login

Παρακάτω φαίνεται το logout page.



#### 4.5 Views

Στο αρχείο `views.py` της εφαρμογής μας υπάρχει όλος ο έλεγχος στο τι γίνεται σε κάθε ένα από τα `template` που έχουμε κατασκευάσει. Για κάθε ένα από τα `template` που έχουμε δημιουργήσει ορίζουμε μια συνάρτηση στην `python` με όρισμα ένα `request`. Κάθε μια από αυτές τις συναρτήσεις υπάρχει για να ελέγχει τα δεδομένα που θα εμφανίζονται στο αντίστοιχο `template`.

Στη κάθε μια συνάρτηση γράφουμε κανονικά κώδικα σε `python` με σκοπό να τραβήξουμε στοιχεία από την βάση δεδομένων που έχουμε, να τα επεξεργαστούμε κατάλληλα και τέλος να επιστρέψουμε στο `template` που επιθυμούμε μεταβλητές που περιέχουν τα δεδομένα που χρειαζόμαστε.

Ακολουθεί ένα μικρό παράδειγμα κώδικα από μια συνάρτηση για τον έλεγχο του `template Account`.

```
def myAccount(request):
    form = MyForm(request.POST or None)
    check_cnl = request.POST.get('cnlBtn')
    if request.method == "POST":
        if form.is_valid():
            if check_cnl:
                Booking.objects.filter(id = int(check_cnl)).delete()
            b_data = Booking.objects.filter(user = request.user)
        return render(request, 'myAccount.html', {"b_data": b_data})
```

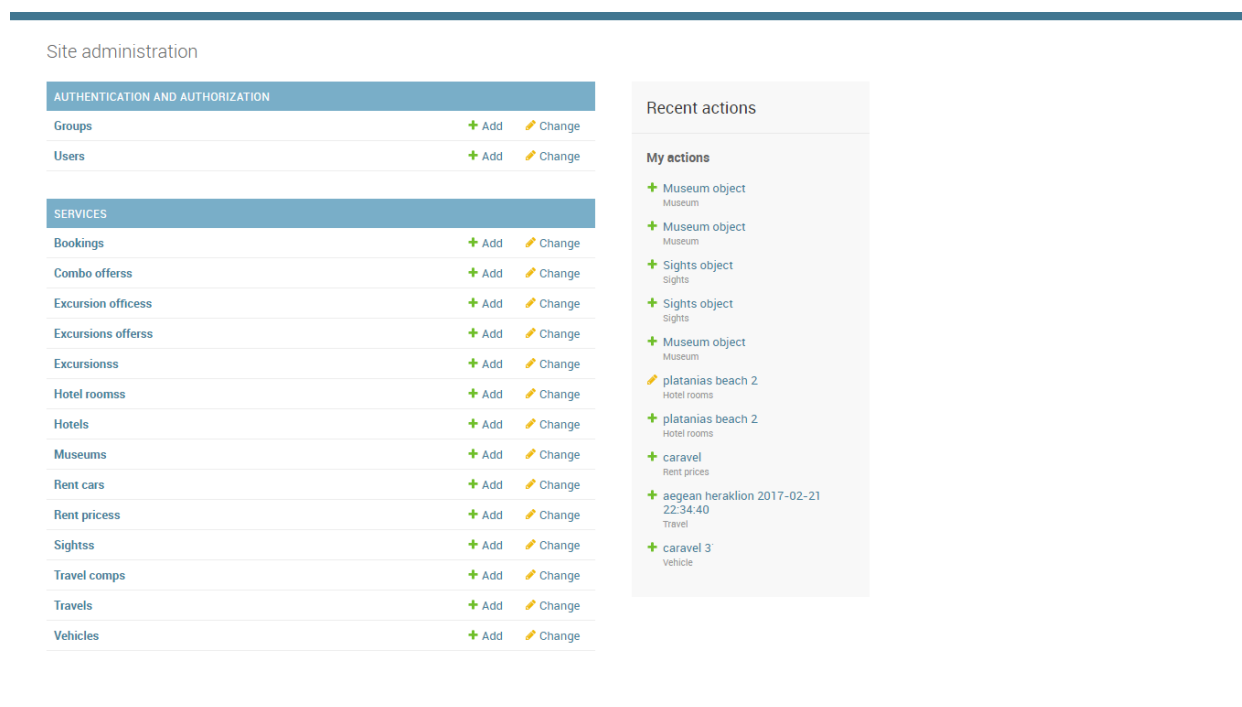
Μετά τη δήλωση της συνάρτησης δηλώνουμε δυο μεταβλητές όπου θα μας βοηθήσουν να χειριστούμε τις διεργασίες που κάνει ο χρήστης και πρέπει να επεξεργαστούμε. Στη συνέχεια γίνεται ένας έλεγχος αν χρησιμοποιούνται οι σωστές μέθοδοι για αποστολή δεδομένων από τον χρήστη στον έλεγχο τις σελίδας. Εάν είναι σωστές τότε προχωράει σε μια ενέργεια, σε αυτή τη περίπτωση διαγραφής μιας καταχώρησης. Τέλος στη μεταβλητή `b_data` αποθηκεύει όλα τα στοιχεία που έχει κάνει `book` ένας συγκεκριμένος χρήστης και τα επιστρέφει στο `template` μέσω αυτής της μεταβλητής.



#### 4.6 Σελίδα Admin

Η σελίδα του admin προσφέρετε για την διαχείριση και την εκχώρηση νέων δεδομένων στην ιστοσελίδα. Πρόσβαση σε αυτή έχει μόνο ο διαχειριστής ή διαχειριστές της ιστοσελίδας. Την σελίδα του Admin την προσφέρει το Django, και αυτό είναι μια από τις μεγάλες ευκολίες που προσφέρει το Framework. Ότι μοντέλα δηλώσουμε στον κώδικα, τα όποια είναι τα μοντέλα που θα αποθηκεύονται στη βάση, το Django τα προσθέτει σε πίνακες στην σελίδα διαχείρισης. Πέρα από αυτούς τους πίνακες από μόνο του προσφέρει την πιστοποίηση και την εξουσιοδότηση ( Authentication and Authorization ) των χρηστών της ιστοσελίδας.

Μετά την εκχώρηση των μοντέλων στη βάση η σελίδα διαχείρισης δείχνει όπως την παρακάτω εικόνα:



Φαίνονται όλα τα μοντέλα που έχουμε δημιουργήσει, καθώς και η δυνατότητα να προσθέσουμε πληροφορία σε αυτά η να αλλάξουμε κάποια ήδη υπάρχον.

Πατώντας πάνω σε ένα από τα μοντέλα που υπάρχουν στη σελίδα θα παρουσιαστούν πληροφορίες και τα δεδομένα που υπάρχουν για αυτό. Ακολουθεί μια εικόνα που δείχνει πως φαίνεται:

Select hotel to change

ADD HOTEL +

Action:   0 of 2 selected

<input type="checkbox"/>	NAME
<input type="checkbox"/>	platanias beach
<input type="checkbox"/>	minoan palace

2 hotels

Το τι πληροφορία θα φαίνεται στον πίνακα περιγράψαμε στο προηγούμενο κεφάλαιο 4.3.

Στη συνέχεια υπάρχει η δυνατότητα προθήκης νέας πληροφορίας – καταχώρησης, με την επιλογή add Hotel όπως φαίνεται στην παραπάνω εικόνα. Αφού επιλέξουμε το add Hotel, θα μας πάει σε μια νέα σελίδα όπου θα μπορούμε να εκχωρήσουμε νέα ξενοδοχεία για την ιστοσελίδα μας. Στη σελίδα αυτή μας ζητάει να δώσουμε πληροφορίες για την νέα καταχώρηση σύμφωνα με τα στοιχεία που έχουμε δηλώσει στα μοντέλα της ιστοσελίδας μας ότι περιέχουν.

Παρακάτω υπάρχει ένα παράδειγμα εκχώρησης ενός νέου ξενοδοχείου. Εκεί ζητάει να εκχωρηθούν τιμές όπως όνομα, διεύθυνση, αριθμός δωματίων, αριθμός τηλεφώνου, καθώς και μια προκαθορισμένη σε επιλογές τιμή ( enum ) που αφορά την κατηγορία του ξενοδοχείου αυτού.

Add hotel

Name:

Address:

Category:



Room num:

Phone num:

[Save and add another](#) [Save and continue editing](#) [SAVE](#)

Παρακάτω άλλο ένα παράδειγμα εκχώρησης θα δοθεί, αυτή τη φορά τα δωμάτια των ξενοδοχείων. Όταν επιλέξουμε προσθήκη νέου θα μας δώσει να συμπληρώσουμε πληροφορία όπως φαίνεται στην παρακάτω εικόνα:

Add hotel rooms

Hotel name:   

Room category:

Room cat num:

Price:

[Save and add another](#) [Save and continue editing](#) [SAVE](#)

Σε αυτό όπως και στο προηγούμενο παράδειγμα προσφέρονται οι ίδιες επιλογές. Εδώ συγκεκριμένα έχουμε την τιμή του δωματίου, το πόσα δωμάτια αυτής της κατηγορίας υπάρχουν, μια προκαθορισμένη σε επιλογές τιμή που αφορά την κατηγορία δωματίων που εισάγουμε. Υπάρχει όμως άλλο ένα πεδίο με προκαθορισμένες επιλογές, το οποίο είναι το ξενοδοχείο που ανήκουν τα δωμάτια που κάνουμε εκχώρηση.

Σε αυτό το πεδίο όπως φαίνεται και από την εικόνα παραπάνω έχουμε την δυνατότητα να κάνουμε νέα εκχώρηση, να αλλάξουμε κάποια ήδη υπάρχον, ή απλά να επιλέξουμε μια από της καταχωρήσεις που υπάρχουν. Μια εικόνα με ανοιχτό το μενού επιλογής δίνεται παρακάτω:

The screenshot shows the Django administration interface for adding hotel rooms. The page title is "Django administration" and the user is logged in as "WELCOME, KORSIK". The breadcrumb trail is "Home > Services > Hotel rooms > Add hotel rooms". The form is titled "Add hotel rooms" and contains the following fields:

- Hotel name:** A dropdown menu with a search icon and a plus sign. The selected option is "minoan palace".
- Room category:** A dropdown menu with two options: "minoan palace" and "platanias beach".
- Room cat num:** A text input field with the value "0".
- Price:** A text input field with the value "0".

At the bottom right of the form, there are three buttons: "Save and add another", "Save and continue editing", and "SAVE".

127.0.0.1:8000/admin/services/hotel/add/?\_to\_field=id&popup=1

## 5. Αποτελέσματα

### 5.1 Συμπεράσματα

Με την ιστοσελίδα αυτή μπορεί να μπει οποιασδήποτε επιθυμεί να επισκεφτεί την Κρήτη και να πάρει πληροφορίες. Μέσα από τη σελίδα μπορεί να κάνει κράτηση σε δωμάτια όπως και σε άλλες υπηρεσίες που προσφέρονται. Μπορεί επίσης να πάρει πληροφορίες για εκδρομές και μέρη που υπάρχουν στην Κρήτη. Δίνεται η δυνατότητα για ειδικά πακέτα προσφορών ώστε να είναι πιο οικονομικά από το να γίνει η κράτηση ξεχωριστά για την κάθε υπηρεσία που επιθυμεί ο χρήστης.

### 5.2 Επεκτάσεις

Οι παραπάνω υπηρεσίες θα ήταν καλό να προστεθούν μελλοντικά για κάθε ανάγκη του χρήστη ώστε να μπορεί να τα βρει όλα από την ιστοσελίδα αυτή. Υπηρεσίες όπως κράτηση σε ταξί καθώς τα δρομολόγια και τις ώρες των λεωφορείων. Έτσι ο χρήστης θα έχει τη δυνατότητα εάν δεν κάνει κράτηση σε κάποιο μεταφορικό μέσο να μπορεί να βρει πως να πάει στους προορισμούς που έχει επιλέξει να επισκεφτεί με άλλους τρόπους. Η δυνατότητα παρουσίασης φωτογραφιών στα δωμάτια των ξενοδοχείων όπως και στις υπόλοιπες υπηρεσίες θα ήταν χρήσιμη επίσης να προστεθεί μελλοντικά. Ένα ακόμα σχέδιο για μελλοντική εργασία στην ιστοσελίδα αυτή είναι να αποστέλλονται πακέτα προσφορές σε χρήστες για περιορισμένο χρόνο. Σε αυτά τα πακέτα θα υπάρχουν υπηρεσίες σύμφωνα με τις προτιμήσεις που έχει δείξει ότι έχει ο χρήστης. Σε ένα μικρό χρονικό περιθώριο η προσφορά αυτή θα ισχύει μόνο για συγκεκριμένο χρήστη και θα έχει πολύ καλύτερες τιμές από τις συνηθισμένες προσφορές.

## Βιβλιογραφία

1. <https://www.djangoproject.com/>
2. <https://www.djangoproject.com/start/overview/>
3. [https://wiki.postgresql.org/wiki/Psycopg2\\_Tutorial](https://wiki.postgresql.org/wiki/Psycopg2_Tutorial)
4. <http://initd.org/psycopg/>
5. [https://en.wikipedia.org/wiki/Python\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))
6. [https://en.wikipedia.org/wiki/Django\\_\(web\\_framework\)](https://en.wikipedia.org/wiki/Django_(web_framework))
7. <https://www.postgresql.org/>
8. <https://el.wikipedia.org/wiki/PostgreSQL>
9. <https://en.wikipedia.org/wiki/SQL>
10. <https://pypi.python.org/pypi/pip>
11. <https://docs.djangoproject.com/el/1.11/howto/windows/>
12. <https://www.cup.gr/book/python/>

### **Python Εισαγωγή στους υπολογιστές. 3η αναθεωρημένη έκδοση**

Συγγραφείς: ΑΒΟΥΡΗΣ ΝΙΚΟΛΑΟΣ, ΚΟΥΚΙΑΣ ΜΙΧΑΗΛ, ΠΑΛΙΟΥΡΑΣ ΒΑΣΙΛΗΣ, ΓΑΡΜΠΙΑΣ ΚΥΡΙΑΚΟΣ

13. <https://www.mgiurdas.gr/biblia/systimata-vaseon-dedomenon-6i-ekdosi>

### **Συστήματα Βάσεων Δεδομένων 6η Έκδοση**

Συγγραφείς: Korth Henry, Silberschatz Abraham, Sudarshan S.

14. <http://www.rosili.gr/el/kainotomia-kai-allagi-sto-epixeirein>

**Καινοτομία και αλλαγή στο επιχειρείν**

Ελένη Ε. Σαλαβού, 2013, Εκδοτικός Οίκος Rosili, ISBN: 978-960-7745-31-6

15. <http://pythonbooks.revolunet.com/>

16. <https://www.postgresql.org/docs/books/>

17. [http://python.org.gr/phocadownload/Tutorials/tutorial\\_by\\_example.pdf](http://python.org.gr/phocadownload/Tutorials/tutorial_by_example.pdf)

18. [http://dide.flo.sch.gr/Plinet/Meetings/Meeting23/A\\_Byte\\_of\\_Python-el.pdf](http://dide.flo.sch.gr/Plinet/Meetings/Meeting23/A_Byte_of_Python-el.pdf)

19. [http://aggelid.mysch.gr/pythonbook/INTRODUCTION TO COMPUTER PROGRAMMING WITH PYTHON.pdf](http://aggelid.mysch.gr/pythonbook/INTRODUCTION_TO_COMPUTER_PROGRAMMING_WITH_PYTHON.pdf)

20. Πανεπιστήμιο Πειραιώς, Τμήμα Πληροφορικής,  
<http://infolab.cs.unipi.gr/pre-eclass/courses/gis/lab/Lab-PostGIS-doc.pdf>

21. [https://repository.ellak.gr/ellak/bitstream/11087/1441/1/postgis\\_qgis\\_gr\\_2.0.pdf](https://repository.ellak.gr/ellak/bitstream/11087/1441/1/postgis_qgis_gr_2.0.pdf)