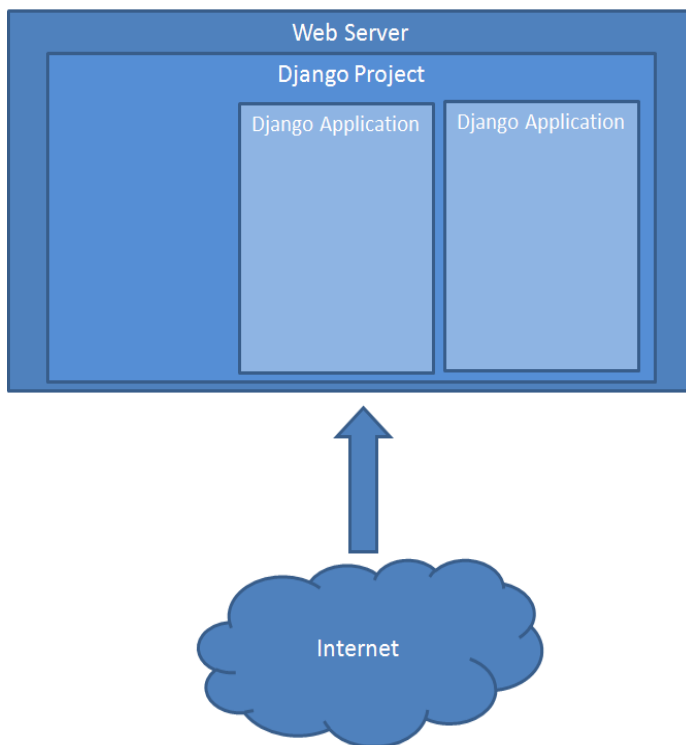
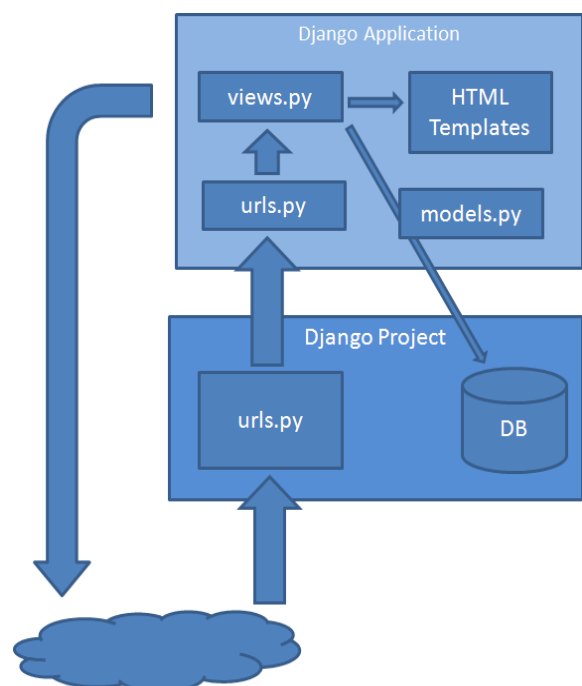


## MEMORIA PRÁCTICA 2

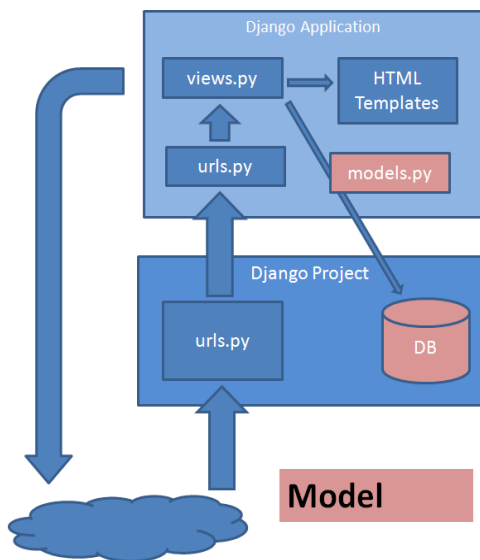


Esta imagen muestra un esquema de la arquitectura de un servidor web según propone Django. El servidor web sería el proyecto Django que contiene las distintas aplicaciones web que ofrecen servicios. También le pertenecen al proyecto la url raíz del mismo y los html base (que pueden ser usados como templates) que serán servidos. En caso de que el proyecto requiera una base de datos, se debe sincronizar una con los modelos descritos en el proyecto. El proyecto también cuenta con un parseador de urls que está sincronizado con el parseador de urls de cada aplicación. La idea es que los html's que servirán las aplicaciones deben cumplir con los templates que proporciona el proyecto.

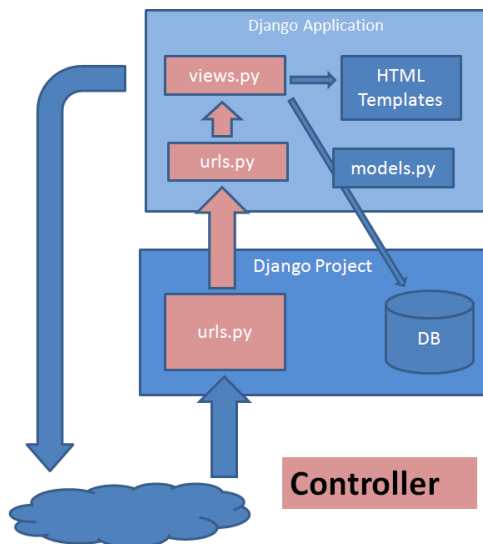
Esta imagen muestra la arquitectura básica de una aplicación Django, de un proyecto Django y de cómo interaccionan entre sí para procesar las solicitudes que llegan. Una petición llega al servidor web y éste le transfiere el control al proyecto Django. Primero, el módulo `urls.py` analiza la url y decide qué función de `views.py` de la aplicación se encargará de presentar una vista. Estas funciones se encargan de preparar un documento html utilizando templates para devolverlo al cliente. Por otra parte, el fichero `models.py` contiene las clases que describen los datos guardados en la tabla de la base de datos relacional. Django permite utilizar varias bases de datos distintas e incluso reutilizar bases de datos existentes.



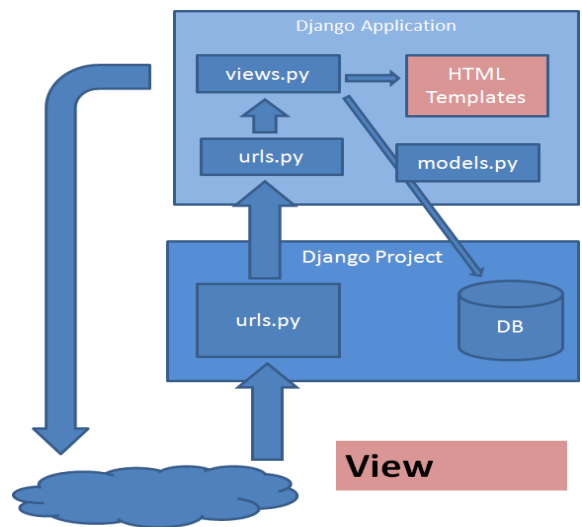
Hecho por: Javier Gómez y Carlos Li



La parte de la vista se encarga de la presentación de documentos html utilizando templates.



Esta imagen hace énfasis en ilustrar que las clases de `models.py` deben describir la estructura de los datos almacenados en la base de datos. Esto significa que los atributos de las clases de `models.py` deben estar reflejados en los datos de la base de datos. De manera que se pueda almacenar la información de una instancia de una clase en la base de datos y se pueda recuperar dicha información de esta instancia.



Por último, esta imagen ilustra la parte del controlador del proyecto, que se encarga de interpretar las urls y de procesar las vistas (para rellenarlas con datos de la aplicación si fuera necesario)