Objetivos:

Los objetivos es ser capaces de crear el código para las operaciones listar (filtar, ordenar…), crear, editar y borrar principalmente. Si mas adelante vemos mas necesidades las vamos incorporando.

Los tipos de datos que se podrán utilizar serán muy diversos: String, text, email, password, boolean int, float, double, porcent (porcentaje), money (moneda), date, time, Object (cualquier ,otra clase). Para seleccionar este último tipo se podrán utilizar select, multiselect o combo. Ya iremos definiendo las implementaciones y si añadimos mas.

Servidor

La estructura que se creará en el servidor tendrá cuatro partes: controller, library, model y views. Para explicar la estructura de los ficheros usaremos la clase Paciente.

* Controllers:
  + pacientes\_base (extends Controller): Este será el fichero en el cual se generará la interfaz de todas las operaciones creadas por el generador de código.
  + pacientes (extends pacientes\_base): Este fichero será donde el usuario podrá definir sus propias operaciones. Esto será así para que una vez creada una versión pueda volver a generar código sin pisar los cambios que haya desarrollado.
* Library:
  + ServicePacientes\_base: Este fichero será el encargado de enlazar el controller con el model, realizando tantas peticiones al model como haga falta para satisfacer las peticiones del controller (pacientes\_base).
  + ServicePacientes (extends ServicePacientes\_base): Idem caso anterior.
* Model:
  + Paciente: (extends Model): El fichero encargado de realizar las operaciones con la base de datos. Mas adelante se definirán las operaciones que irán en el model.
* Views: Esta sección muestra los resultados de las operaciones. Son varios ficheros los que se crean: formulario de creación, formulario de edición, lista en formato json, enum en json…