Big Project

Worth 60% of overall marks
(This is marked out of 100)

What is being assessed (Mark breakdown):

Description	Percent
• **An understanding of REST API:**	
○ **Creating an REST API,**	10%
○ **Writing the server.**	15%
• **Consuming a REST API:**	
○ **Writing a program in either python or JavaScript that consumes an API, either your own API or a third party.**	15%
• **Creating a web interface.**	10%
Sub-Total	50%
• **Elaborating on one or more of these areas**	50%
Total	100%

Assessment strategy:

There are marks (50%) for each of the areas (basic) and optional extra marks (50%) for elaborating on any of the areas.
You have flexibility as to what you do for this project. I understand that this can cause confusion as to what you should do, so here is a table of indicative grade ranges, for the kind of project you hand up.

	Description	Range
A. Web application project		
1	A rehash of the sample project lab with your own data (Week09). I.e.: 1. A basic Flask server that has a 2. REST API, (to perform CRUD operations) 3. One database table and 4. Accompanying web interface, using AJAX calls, to perform these CRUD operations	40%-50%
2	Same as 1, with more then one database table	45%-60%
3	Same as 2, with authorization (logging in)	50%-65%
4	Same as 3, working very smoothly e.g. User error checking, logs etc. Something you can publish	70% +
Extra	The web page looks nice.	Plus 0- 15%
Extra	A more complicated API.	Plus 0 – 15%
Extra	Linking to some third party API.	Plus 5% - 15%
Extra	If the third party API requires authentication.	Plus 0-10%
B. Third Party API project		
5	1. Linking to a simple third party API, 2. Storing the data in a database, 3. Allowing a user to view the database through a web application and, 4. The web page to view that information.	40%-50%
6	Performing some update function through the API (Create, update, Delete).	45%-60%
7	A fully working application.	60%+
Extra	The same extras as above.	

The project should be well laid out and easy for me to run. Marks may be deducted for poorly formatted code.

Handup:

1. A link to the GitHub repository directory that contains the project.
2. A "ReadMe" file if there is anything (complicated) I need to do to run this code.
3. You do not need to host the server on a cloud hosting site (azure) but if you do, please provide the link.
4. Any other documentation you feel is appropriate.

Deadline:

The official deadline for the project is Friday the 13th December, but I will give an automatic extension to Friday 20th December for anyone who asks for it.
My absolute "drop dead" deadline is Tuesday the 7th January, but that is if you are really, really stuck!

Best of Luck.

Email me if you have any questions