**HOMEWORK 1 – M. FACCHINI 547348**

**BEDROCK A PARTY**

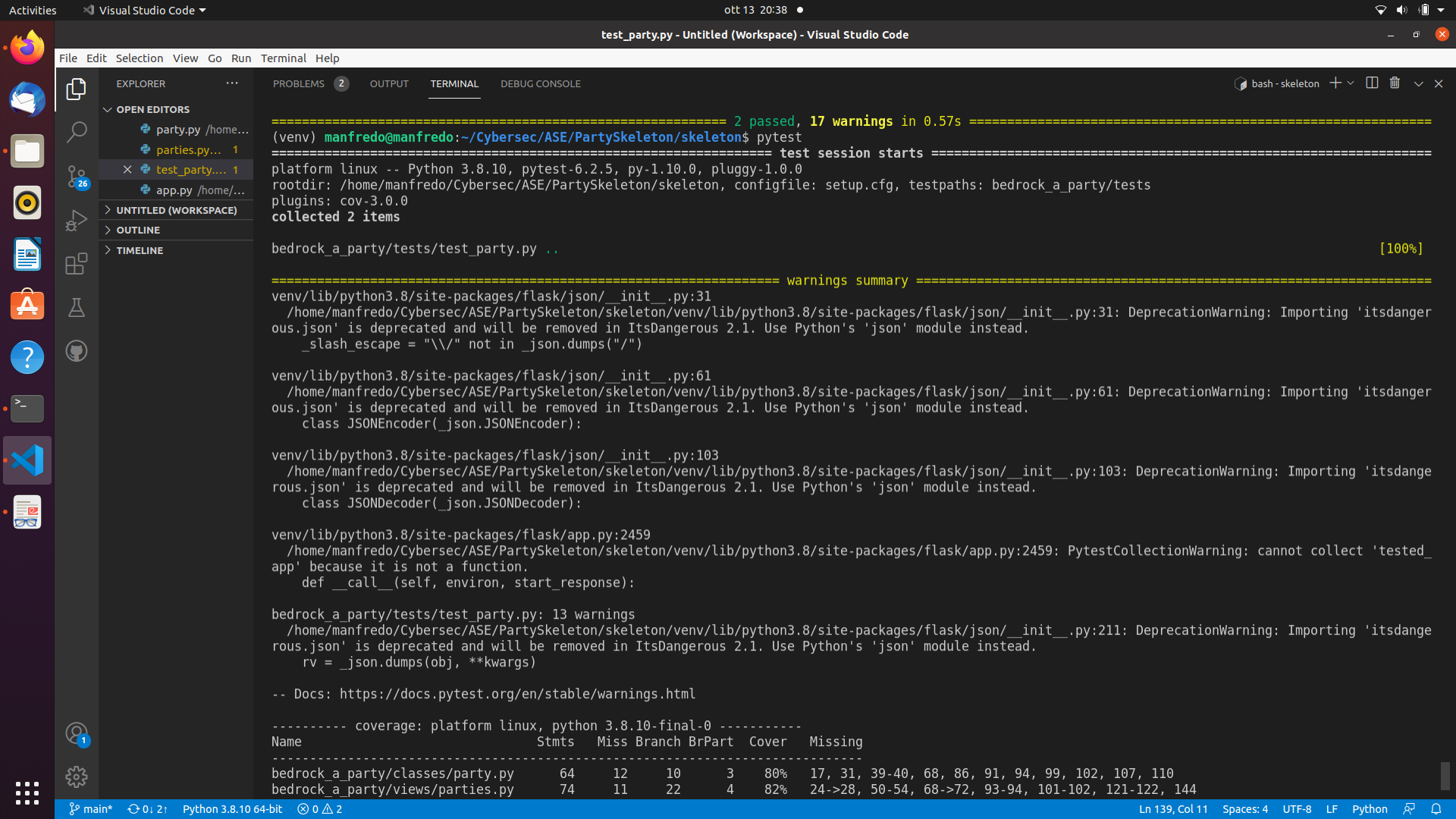
For this first homework, about microservices, I modified routes in /bedrock\_a\_party/views/parties.py file as follows:

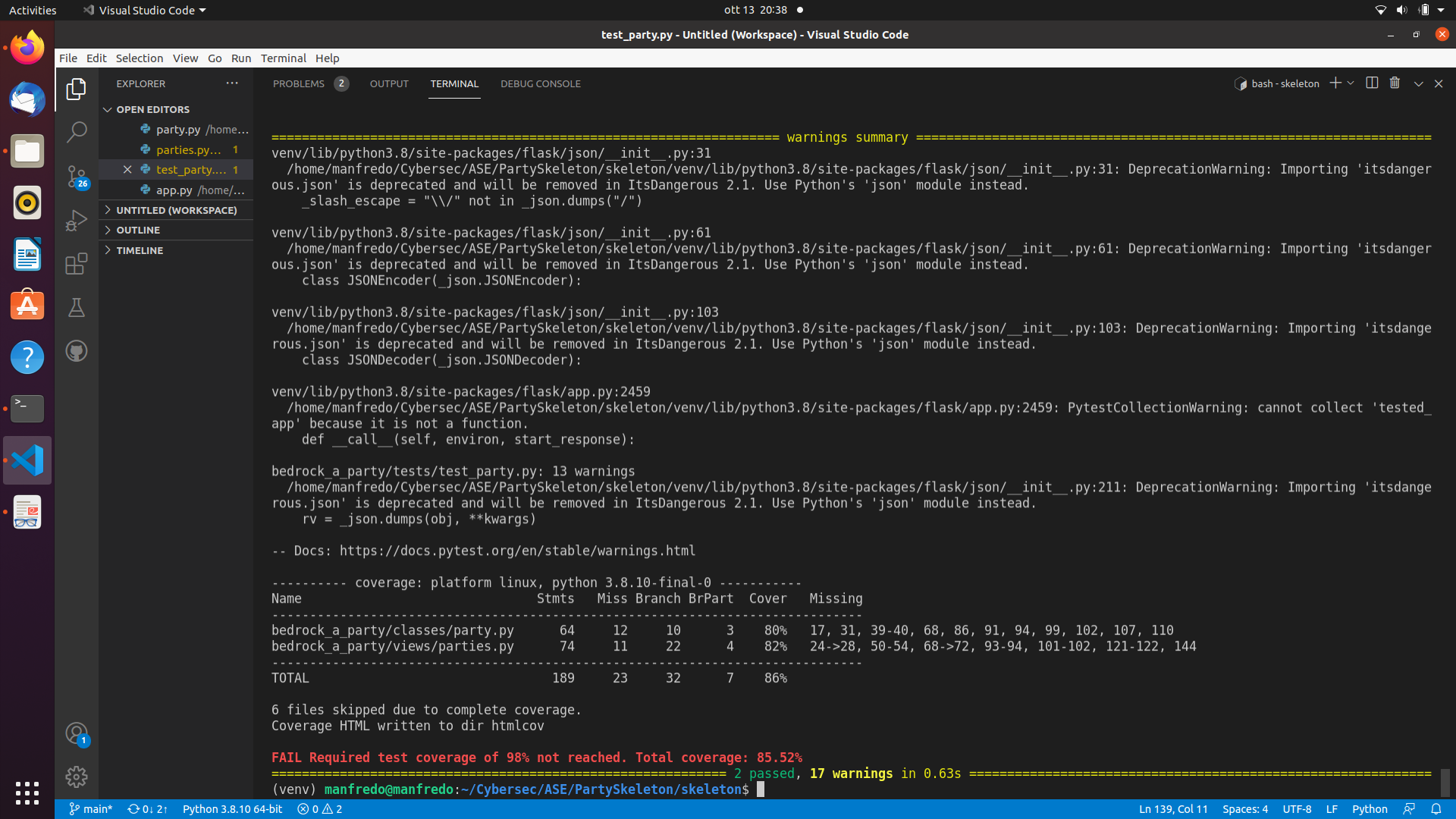
* **/parties**: in case of ‘POST’ method, an instance of Party class is created with the utility function ‘*create\_party(req)’;* in ‘GET’ case all parties are returned with *‘get\_all\_parties()’* function.
* **/parties/loaded**: ‘GET’ method returns the number of parties saved in *\_PARTY\_NUMBER* global variable.
* **/party/<id>**: ‘GET’ method returns the party with identifier *id* using *\_LOADED\_PARTIES* dictionary; ‘DELETE’ method eliminates the instance of Party class with identifier *id*.
* **/party/<id>/foodlist**: the only method ‘GET’ returns the food list of the party, which is identified by *id*, after checking the party’s existence with “exists\_party*(id*)” function*.*
* **/party/<id>/foodlist/<user>/<item>**: ‘POST’ method allows *user* to add a food in the *id* party food list, after checking if the *user* is a party guest and the *item* is not already present in the list; ‘DELETE’ method, after checking the presence of *item* in the food list, allows *user* to remove *item* in party food list.

This report and “parties.py” file could be found at: <https://github.com/ManfredoFac/ASE_facchini.git>

**TEST:**

1. These screenshots represent the tests running with command ‘pytest’ in the IDE terminal (VSCode):





1. These thirteen screenshots, taken with Postman app, represent responses at all requests to the service in order present in /bedrock\_a\_party/tests/test\_party.py file:

