

My Lunch-Web App

Assumptions:

1. It will have to communicate with the restaurants ordering web app/system, to deliver food requests and get responds back, so we need to support diverse data format types.
2. will need security layers, application security, encryption, authentication to secure from web threats/cyber Attacks.
3. The deadline for project delivery is short, so I need a fast programming language and framework (less code and configuration) that can give me response for requests.

Decisions:

1. I chose Python as a programming language (less code write).
2. I chose Flask as a framework/API to meet the requirements, (the other option was to use Django Framework, which requires a lot of time setting configuration).
3. To get more security features I added Flask-Security (micro framework) to quickly add common security mechanisms.
4. I added Flask-RESTful (an expansion for Flask) to support requests:
 - a. different data format types,
 - b. application security-RESTful API can be parametrized/configured securely.
 - c. Flask-RESTful for support ORM for different Data bases
 - d. ORM can all so helps to distinguish data from code, application security for Injections (SQL, OS and etc).
5. Runn the Application Over HTTPS – to get encryption login sessions (PKI) with user.