

PERSONAL DETAILS

Name: Corfu Alin Constantin
Phone no: +4 0753564888
Email address: corfu.alinconstantin@yahoo.ro
corfualinconstantin@gmail.com
Location: Cluj-Napoca, Romania
CV version: 1.06

EDUCATION AND TRAINING

- Qualifications**
- 2009-2013 High School - National College „Calistrat Hogaș” European School - Piatra Neamț
 - 2013-2014 Faculty - Engineering and Management in Public Food and Agrotourism - USAMV – Bucharest
 - 2014-2017 Faculty – Ba. in Philosophy - „Babes Bolyai” University - Cluj Napoca
 - 2017-2019 Faculty – Ma. in Philosophy - „Babes Bolyai” University - Cluj Napoca
 - 2019 – 2022 (expected)- Faculty – Phd. in Philosophy - „Babes Bolyai” University - Cluj Napoca

Languages

Romanian (Native language)
English (C1-B2-B2-B2-B2)
German(A2-A2-A2-A2-A2)

HIGHLIGHTED APPLIED TECHNICAL SKILLS

Programming Languages: **Java**
Technologies: **Spring Framework, Spring Boot**, Maven, Swagger, JUnit5, Postman.
Development **Intellij IDEA**, Postgres.
Operating Systems: Windows

Principal subjects/occupational skills covered

- ❖ Modal Logic, Propositional Logic, Ethics, Epistemology, Ancient Philosophy, Political Philosophy, Hermeneutics
- ❖ Text Editing, Aesthetics, Medieval Philosophy, Philosophy of Religion, Philosophy of Culture, Paleography

WORK EXPERIENCE

- 2017-2018 Librarian at Carturesti Verona - Bucharest (10 months)
 - 2021- 2022 Research Assistant at „Babes Bolyai” University - Cluj Napoca (6 months) in the project PN-III-P4-ID-PCCF-2016-0064 with the title “Nașterea elitei intelectuale în Europa Centrală. Formarea profesorilor la Universitatea din Viena (1389-1450)”
-

Conferences

2015 - Student Association PhiloHistoRiSs - „Stoicism as a way of life in the 21st century”

2017 – *Teacher for one day* – U.B.B. Cluj Napoca

2019 – International day of Philosophy – *Traducere și interpretare în istoria filosofiei* – Round Table.

2020 – Phd. Conference – *Tratatul De sphaera a magistrului Johannes de Sacrobosco în curricula Universității din Paris a secolului al XIII-lea.*

2021 –Phd. Conference – *Începutul și sfârșitul unei tradiții* – *Tratatul De sphaera* al magistrului Iohannes de Sacrobosco. - via Skype, UBB. 2021.

Publications

2020 - Alin Constantin Corfu, *A Short Modern History of Studying Sacrobosco's “De sphaera”*, Studia Universitatis Babes-Bolyai Philosophia, Vol. 65 (LXV), Special Issue, 2020.

Volunteering

2015 – International Federation Institute of Medieval Studies - U.B.B. Cluj Napoca

2015 – Transilvania International Book Festival (FICT) - Cluj Napoca

2017 – *About the logical structure of manipulation* – U.B.B. - Cluj Napoca

2019 – *Medicine and Philosophy: The longue duree of humoral theory* – International Conferences - U.B.B. - Cluj Napoca

Computer skills and competences

Udemy certificate - Spring & Hibernate for Beginners (expected 2022)

Udemy certificate - Java Programming Masterclass covering Java 11 & Java 17 (expected 2022)

Good knowledge of Microsoft Office TM tools - Digital Competency Certificate

Personal Projects - <https://github.com/CorfuAlin>

WebService_Basic_ToProduction

https://github.com/CorfuAlin/WebService_Basic_ToProduction

I. What I wanted to learn and achieve with this project

I wanted to create a Basic Server with 4 methods (GET, PUT, POST, DELETE) focusing on the Post Method that has behind a (Request and Response Model) that will provide the user to input a firstName, a lastName, an email and a password getting as a response a UserId, his firstName, lastName and email.

As a consequence of the user's input on the first hand I wanted to store his password in the database (for this I used the path from UserRequest to UserResponse throw the DTO, UserEntity, UserServiceImpl, UserService interface and UserRepository making use of the Crud Repository). On the second hand I wanted the email to be unique for every new user that was provided from the Post method so I created a costume 500 error message to handle the duplicate entry's using a method in the Service Repository that the Service Implementation Class used coding an if statement that will throw a new RuntimeException.

II. Dependencies

1. Spring Web
2. Spring Data JPA
3. PostgreSQL Driver

III. Future development

1. creating a generated User Public Id (the field in this application remained testUserId)
2. creating the other 3 methods
3. implementing security using SpringSecurity
4. adding XML and Json Support

FastFood_App

https://github.com/CorfuAlin/FastFood_App

I. What I wanted to learn and achieve with this project

As a result of an Udemy course that offered as challenge to create an application in order to manager the process of selling hamburgers I wanted to create a program that would combine and use all the Java-OOP pillars (Inheritance, Encapsulation, Abstraction and Polymorphism) that I got familiar with.

This application is able to select types of burgers and some of the additional items that are added to the burgers and pricing. I created a base Hamburger (Class), with two other types of hamburgers.

The Hamburger (Class) has the following items.

- Bread roll type, meat and up to 4 additional additions (things like lettuce, tomato, carrot, etc.) that the customer can select to be added to the burger. Each one of these items gets charged an additional price in order to calculate the final price. (base burger with all the additions).

Two extra varieties of Hamburgers (subclasses):

a) HealthyBurger (Class) plus two addition items that can be added.

- The Healthy Burger has 6 items (Additions) in total.

b) DeluxeBurger (Class) comes with chips and drinks as additions.

- No extra additions are allowed.

All 3 classes have a method that can be called anytime to show the base price of the hamburger plus all the additions, each showing the addition name, and addition price, and a grand/final total for the burger (base price + all additions).

Testing Service Layer Code with JUnit 5 & Mockito

- ⊗ This section is destined for future development and it will contain the projects up-mentioned (*WebService_Basic_ToProduction* && *FastFood_App*) with a testing possibility making use of JUnit5 and Mockito.