**Competence Document**

**Viktor Petrov**

**1. Persona**

My name is Viktor Petrov, a 19-year-old student originally from Sofia, Bulgaria’s bustling capital city home to over one million people. I moved into my compact but cozy student room in late September, marking another exciting transition in my journey. After transferring from the American College of Sofia at 16, I completed my education at a specialized programming school in Bulgaria—a unique institution co-founded by the creators of its renowned “learn-at-your-own-pace” digital learning platform. This program provided me with an exceptional technical foundation, one that complemented my earlier academic experiences and prepared me well for challenges like adapting to new environments (a skill I’m grateful to continue honing, especially now as I explore life in a different country).

Following graduation—and after savouring a relaxing summer—I decided to pursue university studies right away. Among institutions like TUe and Saxion, I chose Fontys in Eindhoven for its strong ICT programs, and I’m thrilled to be part of the vibrant community there. Beyond academics, my passions lie in all things tech: I love tackling reverse engineering puzzles, designing hardware, soldering circuits, debugging code, and scripting across languages. Over the past few years, I’ve turned countless ideas into projects—from building a home server from an old office PC to creating a customized LLM for portfolio writing at Fontys. My experiments even include a (rather flawed) video game and a souped-up gaming console—proof that curiosity often trumps perfection!

When not immersed in tech, I unwind with my guitar—a hobby I’ve shared by bringing my old acoustic guitar to campus when I discovered a broken one at Fontys TQ. Music’s a great way to connect with others, and I’m glad others can enjoy it too. Cooking has also become an unexpected adventure since living independently; though I once dabbled in elaborate recipes back home, now I focus on balancing budget-friendly, healthy meals (it’s harder than it looks!).

**2. Context**

**Personal project—Lightweight Linux Subsystem**

* **Project Interest**: This is my own, personal project. I would like to learn more about practically applying containers, as well as more practical scripting with python, bash, PowerShell even. I would also really utilise the capability to develop utilising differently-configured environments at a moment’s notice.
* **Reasoning**: I would like to improve my infrastructure setup skills, especially for very specific applications, in this case that being the custom container image servers for my personal project. Also given the amount of programming, it would be a nice brush-up on my skillset.
* **Collaboration**: I would mainly be getting feedback from Mark and Honza. They are both knowledgeable, but there are more experts available to assist me on the way, be it with the infrastructure part or the programming part, or anything else.

**Group project-**

* **Project Interest**: Our group project is oriented towards using a machine model to assist the users of OpenRemote and its software. I like this task as I will be able to improve my skills with hosting machine models at scale. I would also improve my knowledge in model development.
* **Reasoning**: My assigned task is to deal with the hosting, inference engine, backed set up and general infrastructure tasks. This task is quite interesting to me, as this knowledge will be applicable further in my career at Fontys ICT. The real, genuine application of machine learning will only have significance if a significant number of people use it, day-to-day at scale. My tasks at OpenRemote will be a stepping stone towards obtaining that knowledge.

**Collaboration**: Along two great Dutch teammates, I have ample access to documentation and direct support from OpenRemote. This is not just great to speed-up development by directly interacting with the client but is also a great chance to expand my professional network.

**3. Competence profile**

**Current profile**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Managing | Analysing | Advising | Designing | Realising |  | Future-Oriented Organisation | Investigative problem solving | Personal Leadership | Targeted Interaction |
| User interaction | 1 |  | 1 |  | 1 |  |  |  |  |  |
| Business processes |  | 1 | 1 | 1 |  |  |
| Software |  | 1 |  | 1 | 1 |  |
| Hardware |  | 1 |  | 1 | 1 |  |
| Infrastructure |  | 1 |  | 1 | 1 |  |

**Intended development v1**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Managing | Analysing | Advising | Designing | Realising |  | Future-Oriented Organisation | Investigative problem solving | Personal Leadership | Targeted Interaction |
| User interaction |  |  |  |  |  |  | 1 | 1 | 1 | 1 |
| Business processes |  |  |  |  |  |  |
| Software | 1 | 1 | 1 | 1 | 1 |  |
| Hardware |  | 1 |  | 1 | 1 |  |
| Infrastructure |  | 1 |  | 1 | 1 |  |

**Final development**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Managing | Analysing | Advising | Designing | Realising |  | Future-Oriented Organisation | Investigative problem solving | Personal Leadership | Targeted Interaction |
| User interaction |  |  |  |  |  |  |  |  |  |  |
| Business processes |  |  |  |  |  |  |
| Software |  |  |  |  |  |  |
| Hardware |  |  |  |  |  |  |
| Infrastructure |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| Level 1 | Level 2 | Level 3 |

**4. KPI-table with proof**

|  |  |  |
| --- | --- | --- |
| **KPI** | **Proof** | **Rating** |
| **Infrastructure Managing** Manage and monitor basic components of the ICT infrastructure, including hardware and software, to ensure their availability and proper configuration. Follow established procedures to maintain essential infrastructure operations in a structured and predictable context. | **Assignment**:  Set up infrastructure scalability and customizability up to each user’s requirements  **Assignment**: Use Docker containerisation to increase the portability of the end inference solution |  |
| **Infrastructure Designing**  Design a basic part of the ICT infrastructure to support organisational processes based on clearly defined requirements. Follow established guidelines and known solutions to ensure the availability and configuration of traditional hardware and software infrastructure in a structured and predictable context.. | **Assignment:** Analyze the requirements for different performance, expectations and how implementable solutions are **Assignment**:  Offer local hosting capability with limited user hardware |  |
| **Software Analysis** Perform basic analyses of software development processes, products, and information flows in their interrelationship and context. Work under general guidelines in a structured and predictable context, focusing on well-defined problems with known approaches and solutions. | **Assignment:** [**Personal project vs alternatives research**](https://fhict.instructure.com/courses/14859/assignments/261072) |  |
| **Software Advise** Provide advice on organizing basic processes and information for a new or modified software system. This involves following predetermined guidelines and solutions that are known and predictable. The context is structured, and the problems are well-defined, with a clear approach to solving them. | **Assignment: Provide recommendations on optimizing the Linux subsystem for different use cases, such as development, testing, and production environments.**  **Proof: Create a guideline document with best practices and optimization strategies.** |  |
| **Software design:** Design basic software components based on clearly defined requirements. Follow established guidelines and known solutions to ensure the software can be effectively incorporated into the ICT infrastructure after delivery. The context is structured and predictable, focusing on fundamental software development principles. | **Assignment: Design a set of scripts (Python, Bash) to automate the deployment and management of the Linux subsystem.**  **Proof: Share the scripts in a GitHub repository and include a user manual.** |  |
| **Software realisation:** Create and test basic software components based on a given design. Follow established procedures to ensure the software can be effectively incorporated into the ICT infrastructure after delivery. Work within a structured and predictable context to develop fundamental software functionalities. | **Assignment:** **Create a GitHub repository and utilise it for the project**  **Assignment: Develop and test the automation scripts for the Linux subsystem.**  **Proof: Demonstrate the scripts' functionality through a series of test cases and document the results.** |  |
| **Software manage & control:** Manage and monitor basic software development and deployment processes to ensure they align with predefined requirements. Follow established procedures to maintain essential software functionalities within a structured and predictable context, ensuring smooth incorporation into the ICT infrastructure. | **Assignment: Implement a monitoring solution to track the performance and health of the Linux subsystem.**  **Proof: Set up monitoring tools, possibly provide a dashboard showcasing key metrics.** |  |
| **Hardware Realising**  Learning and being proficient at utilisigreal-world hardware formass-scale use | **Assignment: Integrate the Linux subsystem with various hardware configurations to ensure compatibility and performance.**  **Proof: Test the subsystem on different hardware setups and document the results.** |  |
| **Hardware designing:** Design basic software components that interact with available hardware, based on clearly defined requirements. Follow established guidelines to ensure the software effectively considers the capabilities and limitations of the hardware. The context is structured and predictable, focusing on fundamental hardware interfacing principles. | **Hardware Designing**  **Assignment: Design a networking setup that integrates the host's networking interfaces with the Linux subsystem.**  **Proof: Create a simple design document showing how the networking interfaces will be configured. Include a basic diagram and test the setup to ensure it works, then document the results.** |  |
| **Hardware analysis:** Perform basic analyses of hardware interfacing processes, products, and information flows in their interrelationship and context. Work under general guidelines in a structured and predictable context, focusing on well-defined problems with known approaches and solutions, considering the capabilities and limitations of the available hardware. | **Assignment: Analyze the performance of the Linux subsystem on different hardware platforms.**  **Proof: Provide a detailed report comparing the performance metrics across platforms.** |  |
| **Infrastructure analysis:** Analyze processes, products, and information flows within ICT systems, focusing on traditional hardware infrastructure such as networks. Work under general guidelines in a structured, predictable context where the problem is defined, and the approach and solution are known. | **Assignment:**  **Research about docker’s custom image repositories** |  |
| **Infrastructure design:** Design a basic part of the ICT infrastructure to support organisational processes based on clearly defined requirements. Follow established guidelines and known solutions to ensure the availability and configuration of traditional hardware and software infrastructure in a structured and predictable context. | **Assignment:** **Create a plan and implement Docker container to host a simple website** |  |
| **Infrastructure realisation:** Create and test basic components of the ICT infrastructure, including hardware and software, based on a given design. Follow established procedures to ensure the availability and initial configuration of infrastructure elements. Work within a structured and predictable context to support organisational processes. | **Assignment:** **Submit to GitHub and deploy**  **Proof:**  **Utilise GithubActions as a workflow** |  |
| **Professional Development** | | |
| **User Interaction** I want to generally improve my capabilities of translating high-level and detailed, specific ideas into more widely accessible contexts and vice-versa. I have noticed that explaining technical ideas to non-technical people is a personal weak point of mine. | **Goal:** |  |

**5.2 KPI Matrix**

|  |  |  |
| --- | --- | --- |
|  | Mastered |  |
|  | NotMastered |  |
|  | NotAssessed |  |
| 1/2/3/4/5 | Grade |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 🔵Challenge Debriefing | 🔵CLI tool for code generation | 🔵Research AI | 🔵Project Timeline | 🔵CV voor politie | 🔵Plan for Sprint 1 | 🔵Project Documentation | 🔵Competence Document | 🔵Personal Learning Plan |
| Advise-O1 |  |  |  |  |  |  |  |  |  |
| Advise-S2 |  |  |  |  |  |  |  |  |  |
| Advise-U2 |  |  |  |  |  |  |  |  |  |
| Analysis-O1 |  |  |  |  |  |  |  |  |  |
| Analysis-S2 |  |  |  |  |  |  |  |  |  |
| Analysis-U2 |  |  |  |  |  |  |  |  |  |
| Design-O1 |  |  |  |  |  |  |  |  |  |
| Design-S2 |  |  |  |  |  |  |  |  |  |
| Design-U2 |  |  |  |  |  |  |  |  |  |
| Manage&Control-O1 |  |  |  |  |  |  |  |  |  |
| Manage&Control-S2 |  |  |  |  |  |  |  |  |  |
| Manage&Control-U2 |  |  |  |  |  |  |  |  |  |
| Realise-O1 |  |  |  |  |  |  |  |  |  |
| Realise-S2 |  |  |  |  |  |  |  |  |  |
| Realise-U2 |  |  |  |  |  |  |  |  |  |
| PS-2 |  |  |  |  |  |  |  |  |  |
| PL-2 |  |  |  |  |  |  |  |  |  |

**5. Sprint retrospectives**

**Sprint 1:**

For my first, **What I Learned:**

* Gained valuable insights into Fontys and Open Learning.
* Developed professional skills by learning whom and what to ask.

**Positive improvement with networking:**

* Met many helpful individuals who provided excellent advice.
* Had the opportunity to meet Pier, Marijn, and other developers at OpenRemote's office.

**Tools Implemented:**

* Started seting up docker for specific hardware
* Created a few basic assignments to track progress.

**Personal Challenges:**

* This week was personally chaotic but still productive.
* Going forward, I will focus on better time management and structure to improve professionalism.

**Sprint 2**

**What I Learned:**

* Details about different model inference engines, including differences between their strengths and weaknesses.
* Practical limitations with productions at large, such as OpenRemote’s different resource constraints.

**Positive improvements:**

* Recieved great advice on improving my competence documet.
* Got pointers on some minor previous shortfalls, such as not publishing documents in the right place.
* Recieved tips on the usual flow of ideas, planning, research, writing and a lot more structure related to personal projects.
* Recieved specific advice on documents I have hurriedly skipped over (such as a MoSCoW prioritisation and current alternative software research)

**Tools Implemented:**

* This week I personally took an interest in utilising Microsoft’s online services and offerings
* Learned a lot of shortcuts in Microsoft Word Cloud Edition, to be able to more efficiently write documents (such as this competence document)

**Personal Challenges:**

* Even though I can notice an improvement compared to the previous sprint, I notice I still struggle occasionally with motivation
* So far I have been managing with implementing a stricter schedule, but there are rare occasional mistakes that I later fix up, albeit simple ones

**Sprint 3**

**Sprint 4**

**Sprint 5**

**6. Evaluation and Reflection**

**Responsible**  
- Acts consciously with concern for the greater good with contemplation of relevant approaches.  
- Assesses different interests.  
**Innovativeness**  
- Focuses on renewal, improvement and making new connections.  
- Spots or creates opportunities and seizes them.  
**Resilience**  
- Challenges own ideas.  
- Perseveres in finding a result or solution.

**7. Feedpulse**

