**Annex A**

**A Usability Test Report on Component Implementation and Product Derivation in the use of VariaMos**

**ISO/IEC 25062 Common Industry Format for usability Test Reports**

VariaMos1.1.0.1

**Tested by:** Daniel Correa

October 2018

**Date prepared:** October 28, 2018

**Prepared by:** Daniel Correa

**For:**

Computer Science Research Center (CRI) of Panthéon-Sorbonne University

Raúl Mazo

Address inquiries to: Daniel Correa

**Phone:** +57 (4) 4255350

**Email:** dcorreab@unal.edu.co

**Address:** Carrera 80 No 65-223 - Medellín, Colombia

**1. Executive summary**

Software product lines (SPL) have become a successful, but challenging approach to software reuse. A SPL is a collection of software-intensive systems sharing a common, managed set of characteristics that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way. There are different approaches which deal with a SPL implementation; one of them is called Fragment-oriented programming (FragOP), which is a framework used to design and implement SPL domain components. To fully support FragOP we enhanced a modelling tool called VariaMos. VariaMos has been used in several SPL projects and approaches during recent years; this tool incorporates a language to represent and simulate families of systems and (self) adaptive systems. This report presents a usability test of VariaMos (version 1.1.0.1). The main idea is to test the VariaMos usability to support the FragOP approach.

**VariaMos summary**

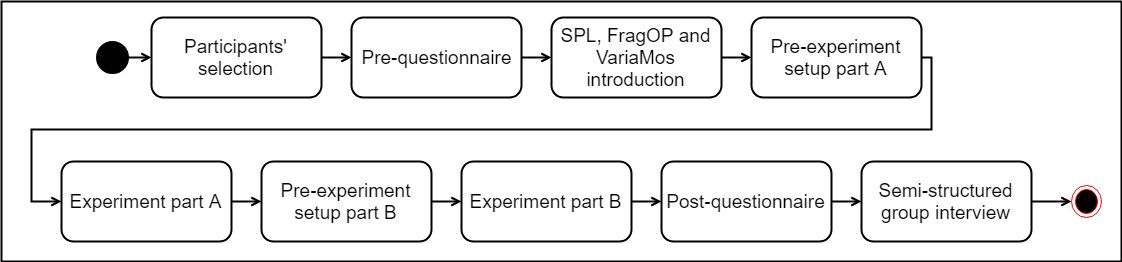
VariaMos is a modeling tool that incorporates a language to represent and simulate families of systems and (self) adaptive systems (Mazo *et al.*, 2015). During the recent years, this tool has been used in several SPL international projects and approaches (Sawyer *et al.*, 2012; Mazo *et al.*, 2012; Mazo *et al.*, 2015; Correa *et al.*, 2018). Different research groups of different countries, such as Colombia and France have been improving this tool.

Currently, VariaMos present some capabilities, such as product line requirements modelling and product simulation which is useful to design and implement a SPL. We took advantage of these capabilities and we improved VariaMos with new capabilities to support the FragOP process: (i) domain components modelling, (ii) binding (or weave) the product line requirements model and the domain components model, (iii) configure new products from the domain models, (iv) derive the configured products, (v) customize the derived products, and (vi) verify the domain models and the derived products. Only one FragOP activity (“implementing domain components”) is not supported by VariaMos and must be carried out with the use of external software (with integrated development environments; IDE).

**Method summary**

The usability test was designed as a process with nine activities (see Figure 1), which is described next.

1. **Participants’ selection:** eight graduate students from the Universidad Nacional de Colombia participated of this test. These students were attending to a postgraduate course in software modelling at the Universidad Nacional de Colombia.
2. **Pre-questionnaire (time 15 minutes):** the second activity was to distribute a pre-questionnaire to the participants. The pre-questionnaire was used to ascertain the participants’ background and general profile. It also served to confirm the students’ lack of knowledge with FragOP and VariaMos. The pre-questionnaire was designed using a Likert scale (Likert, 1932), which had a five-point format: (1) strongly disagree, (2) somewhat disagree, (3) neither agree nor disagree, (4) some-what agree, and (5) strongly agree.
3. **SPL, FragOP and VariaMos introduction (time 3 hours):** the third activity consisted on a magistral class in which we presented a brief presentation of: (i) the main SPL concepts, (ii) the FragOP main concepts, and (iii) the VariaMos tool and an example of a component implementation and product derivation with the use of the FragOP elements.
4. **Pre-experiment setup part A (30 minutes):** the participants were introduced to a document which presented a series of steps to set up a SPL project with the use of VariaMos. Therefore, it showed how to run a derived product in the Eclipse EE environment. This project was the base to carry out the experiment part A.
5. **Experiment part A (90 minutes):** the first part of the experiment included the development of five different tasks. Such as two product derivations, FragOP main concepts understanding, component modification, and fragment creation.
6. **Pre-experiment setup part B (15 minutes):** the participants were introduced to a document which presented a series of steps to set up a SPL project with the use of VariaMos. This project was the base to carry out the experiment part B.
7. **Experiment part B (30 minutes):** the second part of the experiment included the configuration of a new SPL project and the development of two different tasks (finding derivation and validation errors).
8. **Post-questionnaire (15 minutes):** the participants were submitted to the post-questionnaire, which included questions about: (i) experiment environment, (ii) overall satisfaction, (iii) VariaMos and FragOP performance, (iv) general question, and (v) specific questions about the VariaMos and FragOP theory. The post-questionnaire was also designed using a Likert scale.
9. **Semi-structured group interview (25 minutes):** at the end of the experimentation, a semi-structured group interview with all the participants was carried out. We asked the four open questions about the tool usability and we recorded the participant’s answers.



**Figure 1.** Usability test process (Activity diagram UML)

**Summary performance results**

Eight graduate students participated in this study, each participant performed seven tasks. We recorded the assisted task completion rate, unassisted task completion rate, time to task completion, number of assists, number of errors, and questionnaire results. And a subjective post-questionnaire was also administered at the end to obtain qualitative satisfaction data. All participants completed all the seven tasks successfully; the mean task completion was 72.125 minutes for all tasks, and a mean of number of assistances of 0.875 (see Table 1).

**Table 1.** Summary performance results

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Assisted task completion rate** | **Unassisted task completion rate** | **Mean total task completion time** | **Errors** | **Assistances** | **Efficiency** |
| **All participants** | 100 | 100 | 72.125 | 0 | 0.875 | 9.471 |

**2. Introduction**

The next subsections provide a description of VariaMos and the test objectives.

**2.1 Full product description**

**Product name:** VariaMos

**Product Version:** 1.1.0.1

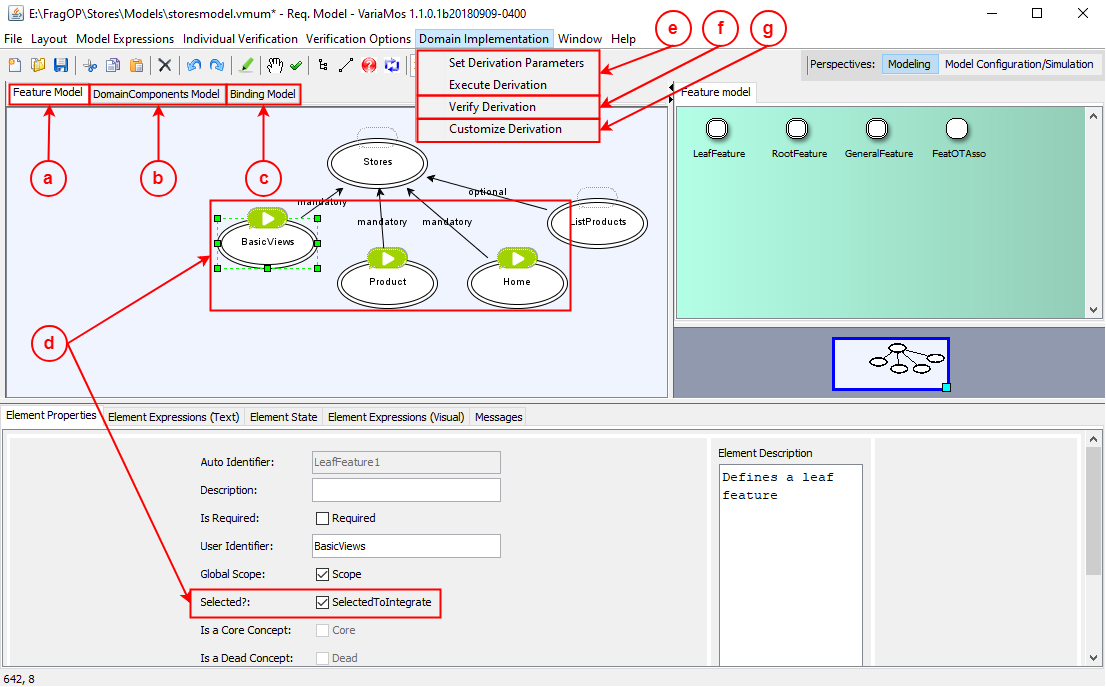
**Website:** https://variamos.com

**GitHub:** <https://github.com/SPLA/VARIAMOS>

**VariaMos target users:** this tool is intended to be use for researchers, students, software developers and industrials which are interested in the software product line methodology. This includes people with experience in the software product lines field, and novice people which are interested in adopting a software product line methodology. Finally, this tool can be also used for people which is interested in the component-based software development.

**VariaMos assistive technologies:** SWI-Prolog, MXGraph.

**VariaMos evaluated parts:** (a) feature modelling, (b) domain component modelling, (c) binding modelling, (d) product configuration, (e) product derivation, (f) product verification, and (g) product customization. Additionally, we evaluated the “implementing domain components” FragOP activity through code checking and some google drive screenshots, but this was not supported by VariaMos.



**Figure 2.** VariaMos evaluated parts

**2.2 Test objectives**

As previously stated, the main idea is to test the VariaMos usability to support the FragOP approach. This is why we focused on seven VariaMos parts as seen in Figure 2. Additionally, we consider the domain component implementation which was not supported by VariaMos. So, we plan to test:

1. **Feature modelling:** the ability of the users to interact and modify the feature models.
2. **Domain component modelling:** the ability of the users to interact and modify the component models.
3. **Binding modelling:** the ability of the users to interact and modify the binding models.
4. **Product configuration:** the ability of the users to interact and create custom product configuration.
5. **Product derivation:** the ability of the users to derive software products and to find product derivation errors.
6. **Product verification:** the ability of the users to verify software products and to find product verification errors.
7. **Product customization:** the ability of the users to customize software products.
8. **Domain component implementation:** the ability of the users to interact, modify and create domain component files.

We also included a subjective satisfaction questionnaire about VariaMos and FragOP (as part of the post-questionnaire).

**3. Method**

The next subsections provide information to allow an independent tester to replicate the procedure used in this testing. Therefore, it provides detailed information about the activities defined in the usability test process (see Figure 1).

**3.1 Participants**

Eight graduate students from the Universidad Nacional de Colombia participated of this testing. These students were attending to a postgraduate course in software modelling at the Universidad Nacional de Colombia. The participants agreed to develop the usability test during two classes of the software modelling course.

Additionally, before the usability test execution, we submitted the participants to a pre-questionnaire. In order to collect information about the participants’ background and experience, and in order to confirm the participants’ lack of knowledge with FragOP and VariaMos (see Table 2 and Table 3). The participants’ background and experience were very important to develop the usability test. The pre-questionnaire results showed that participants have knowledge in software development (mean = 4,000; stdev = 0,535) and more or less knowledge in component development (mean = 3,125; stdev = 0,641). The pre-questionnaire also shows that the participants have lack of knowledge in SPL (mean = 1,250; stdev = 0,463), VariaMos (mean = 1; stdev = 0), and FragOP (mean = 1; stdev = 0). The last two results were mandatory to execute the experiment.

Based on the participants’ background and experience, these participants could be classified as “novice people which are interested in adopting a software product line methodology” which is one of the VariaMos user target population.

**Table 2.** Participants’ pre-questionnaire information summary first part

|  |  |  |  |
| --- | --- | --- | --- |
| **Participant** | **Gender** | **Job occupation** | **Professional experience (years)** |
| P1 | Male | Software Development and Automatization | 3 |
| P2 | Male | Software Development | 7 |
| P3 | Male | Software Development | 3 |
| P4 | Male | Software Development | 3 |
| P5 | Male | Web development | 3 |
| P6 | Male | Software Development and Database Analyst | 5 |
| P7 | Male | Software Development | 8 |
| P8 | Male | Web development | 3 |

**Table 3.** Participants’ pre-questionnaire information summary second part

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Participant** | **Experience with software development** | **Experience with component development** | **Experience with SPL implementation** | **Experience with VariaMos** | **Experience with FragOP project development** |
| P1 | 5 | 3 | 1 | 1 | 1 |
| P2 | 4 | 3 | 2 | 1 | 1 |
| P3 | 3 | 3 | 1 | 1 | 1 |
| P4 | 4 | 4 | 2 | 1 | 1 |
| P5 | 4 | 4 | 1 | 1 | 1 |
| P6 | 4 | 3 | 1 | 1 | 1 |
| P7 | 4 | 2 | 1 | 1 | 1 |
| P8 | 4 | 3 | 1 | 1 | 1 |
| Mean | 4.000 | 3.125 | 1.250 | 1.000 | 1.000 |
| Standard Deviation | 0.535 | 0.641 | 0.463 | 0.000 | 0.000 |
| Standard Error | 0.189 | 0.227 | 0.164 | 0.000 | 0.000 |
| Min | 3 | 2 | 1 | 1 | 1 |
| Max | 5 | 4 | 2 | 1 | 1 |

**3.2 Context of product use in the test**

**3.2.1 Tasks**

* **Description of task scenarios:** there was an experiment with tow parts (A and B). For each part we provided a SPL models and components code (a SPL project). And based on each project, the participants have to develop some tasks (part A contained the development of 5 tasks, and part B the development of 2 tasks).
  + **Part A:** included the development of tasks such as two product derivations, FragOP main concepts understanding, component modification, and fragment creation.
  + **Part B:** included the development of tasks such as finding derivation errors and finding validation errors.
* **Why tasks were selected:** as previously stated, the main idea was to test the VariaMos usability to support the FragOP approach. The previous tasks focused on all the FragOP process inside the VariaMos environment.
* **Task source:** we designed two small SPL projects over VariaMos as the base for the task developments. For these projects we designed some specific task that represent the typical use scenario that VariaMos users will find when they work with this tool.
* **Task data given to participants:** two projects (one for Experiment Part A, and another for Experiment Part B). These projects contained: (i) software required, (ii) pre-developed components, (iii) pre-developed models, and (iv) instructions to run the project.
* **Task performance criteria:** each task has its own completion criteria. Some of these tasks required to answer some questions, other required to put a screenshot, other to upload some pieces of code. At the end of the experiment these tasks were validated. The final score for each task was a result ranging from 0% (not completed) to 100% (completed).

**3.2.2 Test facilities**

* **Intended context of use:** any workplace with people interested in SPL or component-based software development.
* **Context used for this test:** the usability test was conducted in a laboratory of the Facultad de Minas at the Universidad Nacional de Colombia. The participants use their own computers with a list of software programs previously installed and configured. They were requested to record their tasks through the use of google drive documents. Additionally, two test administrators were observing and attending the participants questions. Finally, we use audio record programs to record the participants opinions at the end of the experiment.

**3.2.3 Participants’ computer environment**

* **Intended context of use:** VariaMos 1.1.0.1 is intended to use for any PC running Windows or Unix operating systems.
* **Context used for this test:** as previously stated, we requested the participants to bring their own PC. This way we will test how the tool perform over very different PC configurations. Some of the participant PC included: (i) operating systems such as Windows, Linux, and macOS; (ii) RAM between 2 GB and 12 GB; (iii) display units between 14” and 17”; and (iv) the software programs required to execute the projects. The software programs included: VariaMos 1.1.0.1, Eclipse Oxygen EE 3, Java (JDK 8), Apache Tomcat 8.5.

**3.2.4 Test administrator tools**

* Tasks were recorded with the use of Google Drive documents.
* At the beginning of the experiment the participants were submitted to a pre-questionnaire.
* At the end of the experiment the participants were submitted to a post-questionnaire.
* We collected the participant developed tasks (including code, models and screenshots).
* We also recorded a semi-structured group interview with all the participants.
  1. **Experimental design**

For the experimental design all participants were requested to complete the same experiment divided into two parts. The participants were not divided into groups. The only control variable was the participant ID which was used for statistical purpose. Other individual variables were recorded through the pre-questionnaire such as users experience with software development, component development, VariaMos, and FragOP. However, differences between participants were not analyzed in detail (a summary was presented in Section 3.1).

There were 5 dependent variables: unassisted task completion rate, time to task completion, number of assists, number of errors, and questionnaire results. And a subjective post-questionnaire was also administered at the end to obtain qualitative satisfaction data.

* + 1. **Procedure**

Upon arrival, the participants were introduced to the VariaMos usability test, and they were requested to participate. We told them that we want to find errors, and the evaluation also will help to make the tool easier to learn and use. We told them that their personal information (such as name, email, and company, among others) won’t be exposed, so they will appear in the report as participant 1..N. We explained to them that the usability test was about a software tool used to design and implement SPL components, and they will be requested to complete some tasks. Next, participants were asked to sign a document to participate in the experiment.

Before the experiment execution the participants were requested to complete a pre-questionnaire about their background and software experience (5-point Likert scale). After the pre-questionnaire, we started a magistral class. In this class we presented the main concepts of SPL, FragOP and VariaMos, and we developed a very small example about the use of FragOP and VariaMos.

After the magistral class, the participants were requested to follow a “pre-experiment setup part A”, here, the participants were introduced to a document which presented a series of steps to set up a SPL project with the use of VariaMos. Once all participants completed the pre-experiment setup, we shared with them a personal Google Drive folder with the experiment part A. Then, they started to complete the five experiment part A tasks. There was a limit of 90 minutes for this part. Additionally, two test administrators were observing and attending the participants’ questions.

After the experiment part A, the participants were requested to follow a “pre-experiment setup part B”, here, the participants were introduced to a document which presented a series of steps to set up a SPL project with the use of VariaMos. We share with them the experiment part B document. Then, they started to complete the two experiment part B tasks. There was a limit of 30 minutes for this part. Additionally, two test administrators were observing and attending the participants questions.

Finally, after the experiment part B completion, the participants were requested to submit a post-questionnaire. And at the end, there was a semi-structured group interview about VariaMos and FragOP.

* + 1. **Participant instructions**

All the documents, including the: (i) pre-experiment setup part A, (ii) experiment part A, (iii) pre-experiment setup part B, and (iv) experiment part B; they can be found online here (Correa, 2018).

* 1. **Usability Metrics**
     1. **Effectiveness**
* **Completion rate:** unassisted completion rate was defined as the percentage of participants who completed each task correctly without any assistance from the test administrator. Assisted completion rate was defined as the percentage of participants who completed each task correctly with the test administrator intervention.
* **Errors:** an error was defined as a task completed wrongly or not completed.
* **Assists:** an assist was defined as verbal help given by the test administrators to guide the participant to the next step in completing the task. Test administrators also provide help to participant to understand a task definition that was not very clear to them; but this type of assists was not recorded.
  + 1. **Efficiency**
* **Task time:** the amount of time to complete each task.
* **Completion rate efficiency:** mean completion task rate/mean task time.
  + 1. **Satisfaction**

At the end of the experiment, participants completed a post-questionnaire. Subjects rated some questions on a 5-point Likert scale, and there were other questions about VariaMos and FragOP. Scores were given for each participant's perception of: ease of use, easy to learn, easy to remember and subjective satisfaction. Therefore, a semi-structured group interview was carried out about the tool usability.

**4. Results**

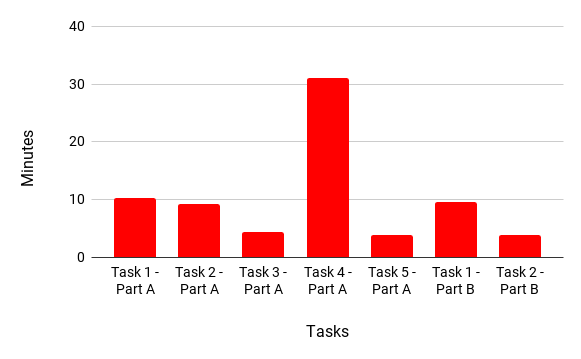
**4.1 Data analysis**

* **Data scoring:** participant behavior was categorized into successes (task completion), errors and assistances. An error was defined as a task completed wrongly or not completed. Likewise, for each time the test administrator felt that the participant needed verbal assistance in order to complete a task, and when the assistance was given, the participant received a score for assistances.
* **Data reduction:** data for each task was analyzed separately and summarized together.
* **Statistical analyses:** descriptive statistics used included: means, standard deviations, standard errors of the means, minimum values and maximum values.
  1. **Presentation of the results**
     1. **Performance results**

All the eight participants completed all the seven tasks. Three of them completed all the seven tasks without assistances. A total of seven assistances were given to the participants, five of these assistances were requested to the Task 4 – Part A, which was the most the complex task (participants spend a mean of 31 minutes to complete this task). The mean total time to completion all the seven tasks was approximately 72 minutes. There were not errors because all the participants completed all the tasks properly (see Table 4).

**Table 4.** Participants’ performance result summary

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Participant** | **Assisted task completion rate** | **Unassisted task completion rate** | **Total Task time** | **Errors** | **Assistances** | **Mean task time** | **Efficiency** |
| P1 | 100 | 100 | 81 | 0 | 2 | 11.571 | 8.642 |
| P2 | 100 | 100 | 72 | 0 | 1 | 10.286 | 9.722 |
| P3 | **-** | 100 | 71 | 0 | 0 | 10.143 | 9.859 |
| P4 | 100 | 100 | 85 | 0 | 1 | 12.143 | 8.235 |
| P5 | **-** | 100 | 61 | 0 | 0 | 8.714 | 11.475 |
| P6 | **-** | 100 | 53 | 0 | 0 | 7.571 | 13.208 |
| P7 | 100 | 100 | 90 | 0 | 2 | 12.857 | 7.778 |
| P8 | 100 | 100 | 64 | 0 | 1 | 10.429 | 6.849 |
| Mean | 100.000 | 100.000 | 72.125 | 0.000 | 0.875 | 10.464 | 9.471 |
| Standard Dev | 0.000 | 0.000 | 12.654 | 0.000 | 0.835 | 1.746 | 2.075 |
| Standard Error | 0.000 | 0.000 | 4.474 | 0.000 | 0.295 | 0.617 | 0.734 |
| Min | 100.000 | 100.000 | 53 | 0 | 0 | 7.571 | 6.849 |
| Max | 100 | 100 | 90 | 0 | 2 | 12.857 | 13.208 |

****

**Figure 2.** Participants’ average time to complete each task

Figure 2 shows that the participants spend more time developing Task 4 – Part A (mean of 31 minutes). This task was about creating a fragmentation point and a fragment. This task involved: (i) the creation of a new feature, (ii) the creation of a new domain component and file, (iii) the development of a fragment and a new fragmentation point (without support of VariaMos), and (iv) the update of the binding model. It also shows that the participants spend little time in the development of the Task 5 – Part A and Task 2 – Part B. Task 5 – Part A was about a new product derivation, which took on average approximately 4 minutes; Task 2 – Part B was about finding validation errors, we included a syntax error over a domain component file and in average the participants only spend approximately 4 minutes in finding and fixing the error.

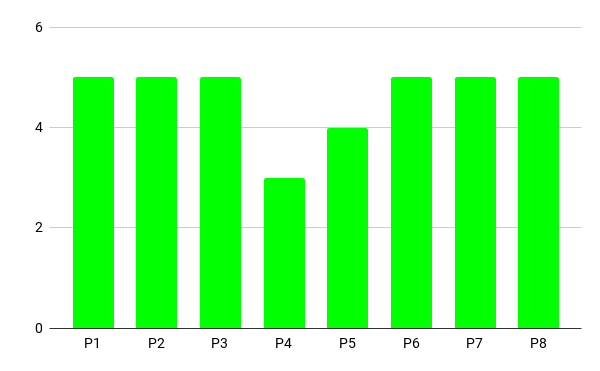
The results of this study show that all the participants were able to complete a set of tasks in which VariaMos supported the FragOP process. Including:

* Feature modelling, through: Task 4 – Part A.
* Domain component modelling, through: Task 4 – Part A.
* Binding modelling, through: Task 4 – Part A.
* Product configuration, through: Task 1 – Part A, Task 5 – Part A.
* Product derivation, through: Task 1 – Part A, Task 1 – Part B, Task 5 – Part A.
* Product verification, through: Task 2 – Part B.
* Product customization, through: Task 1 – Part A.
* Domain component implementation, through: Task 2 – Part A, Task 3 – Part A, Task 4 – Part A, Task 1 – Part B, Task 2 – Part B.
  + 1. **Satisfaction results**

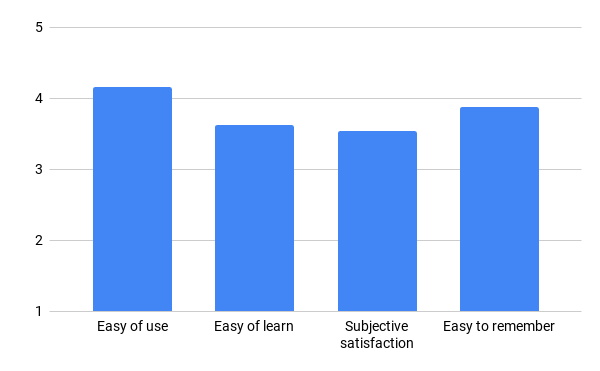
The participants were asked to complete a post-questionnaire about the VariaMos usability to support the FragOP process. The summary results can be seen in Table 5. The highest satisfaction result was about the “ease of use” of VariaMos with a mean of 4.153 (see Figure 4). Therefore, in average the participants had 4.6 correct answers of a total of 6 when asked about VariaMos and FragOP functionalities (see Figure 3).

**Table 5.** Participants’ satisfaction result summary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Participant** | **Ease of use** | **Easy of learn** | **Subjective satisfaction** | **Easy to remember** | **VariaMos and FragOP correct answers** |
| P1 | 4.2 | 4.0 | 2.5 | 5.0 | 5.0 |
| P2 | 4.1 | 5.0 | 5.0 | 4.0 | 5.0 |
| P3 | 3.9 | 4.0 | 4.0 | 4.0 | 5.0 |
| P4 | 4.0 | 4.0 | 3.5 | 3.0 | 3.0 |
| P5 | 5.0 | 1.0 | 3.8 | 4.0 | 4.0 |
| P6 | 4.6 | 5.0 | 3.3 | 4.0 | 5.0 |
| P7 | 3.8 | 4.0 | 3.5 | 4.0 | 5.0 |
| P8 | 3.7 | 2.0 | 2.8 | 3.0 | 5.0 |
| Mean | 4.153 | 3.625 | 3.531 | 3.875 | 4.625 |
| Standard Dev | 0.440 | 1.408 | 0.773 | 0.641 | 0.744 |
| Standard Error | 0.156 | 0.498 | 0.273 | 0.227 | 0.263 |
| Min | 3.7 | 1.0 | 2.5 | 3.0 | 3.0 |
| Max | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 |

****

**Figure 3.** Participants’ correct answers about VariaMos and FragOP

****

**Figure 4.** Satisfaction question average results

Finally, the semi-structured interview showed that in general participants liked the software application and saw the potential of this tool and the FragOP approach. There were some recommendations to improve the tool, about the automatically generation of the component model based on the component pool folder information, and about the way to display some errors.

**Appendix**

* **Program installation:** <https://github.com/danielgara/FragOP-thesis/blob/master/Usability%20test/Program%20installation.docx>
* **Pre-experiment setup A:** <https://github.com/danielgara/FragOP-thesis/blob/master/Usability%20test/Pre-experiment%20setup%20A.docx>
* **Experiment Part A:** <https://github.com/danielgara/FragOP-thesis/blob/master/Usability%20test/Experiment%20Part%20A%20-%20UNAL.docx>
* **Pre-experiment setup B:** <https://github.com/danielgara/FragOP-thesis/blob/master/Usability%20test/Pre-experiment%20setup%20B.docx>
* **Experiment Part B:** <https://github.com/danielgara/FragOP-thesis/blob/master/Usability%20test/Experiment%20Part%20B%20-%20UNAL.docx>
* **Pre-questionnaire:** <https://github.com/danielgara/FragOP-thesis/blob/master/Usability%20test/Pre-questionnaire.docx>
* **Post-questionnaire:** <https://github.com/danielgara/FragOP-thesis/blob/master/Usability%20test/Pos-questionnaire.docx>

**References**

**Correa, D. (2018). FragOP-Thesis GitHub repository, Available at: https://github.com/danielgara/FragOP-thesis**

**Correa, D., Mazo, R., & Giraldo-Goméz, G.L. (2018). Fragment-oriented programming: a framework to design and implement software product line domain components. DYNA, vol. 85(207), pp. 74-83.**

**Likert, R. (1932). A technique for the measurement of attitudes. Archives of psychology.**

**Mazo, R., Muñoz-Fernández, J. C., Rincón, L., Salinesi, C. & Tamura, G. (2015). VariaMos: an extensible tool for engineering (dynamic) product lines. In: Proc. of the 19th Int. Conf. on Software Product Line, pp. 374-379, ACM.**

**Mazo, R., Salinesi, C., & Diaz, D. (2012). VariaMos: a tool for product line driven systems engineering with a constraint based approach. In 24th International Conference on Advanced Information Systems Engineering (CAiSE Forum'12).**

Sawyer, P., Mazo, R., Diaz, D., Salinesi, C., & Hughes, D. (2012). Using constraint programming to manage configurations in self-adaptive systems. Computer, vol. 45(10), pp. 56-63.