## Developer Competition Evaluations

INESC-ID and AlticeLabs organized a developer competition in Lisbon, which took place on Saturday, May 27th, 2017. The event was promoted via [Meetup](https://www.meetup.com/Decentralised-Comms-LX/events/239818330/). It started at 9:30 and lasted until 19:00, taking place at the Techtris House, a startup incubator and home to the WebSummit offices in Lisbon. The goals of the competition were to promote the ReThink platform and gather developer feedback. The competition had developers, who were completely unfamiliar with the ReThink platform, complete a set of 5 challenges. Challenges started with creating a simple application that made use of a Hyperty, progressing through combining multiple hyperties, creating a new hyperty and concluding with the development of a video-conferencing web application making use of available hyperties. At the end of each challenge, developers were asked to fill a questionnaire about the task they had just completed. The competition had 12 participants, grouped into 8 teams of at most two elements. Most participants were male, but a female attended. Participants’ ages ranged from 20 to 70 years. Two groups made significant progress into the fourth challenge. A jury made up of two INESC-ID and two Altice Labs representatives awarded the first prize (worth 500€) to Diogo Pereira. Participation awards were given to all participants. Overall, participants recognized value to the ReThink platform, saying they would recommend its use. Some of the suggestions included:

* Offer built-in integration with front-end frameworks for UI creation
* Having a larger offer of hyperties for developers to integrate into their apps

The most valuable features were found to be:

* Ease of integrating hyperties for complex tasks into new applications
* Ease of use
* The ability to interoperate with applications from other developers by reusing their hyperties

These conclusion were reached though the use of the questionnaire presented in this section, that was used to evaluate different aspects of the reTHINK system and its development framework. The main goal of this questionnaire was to collect feedback on the reTHINK Beta Tester Event, organized on May 27th for a limited number of outside developers. This questionnaire was divided into seven sections. One section for team caracterization and background, five sections corresponding to each of the challenges and a last section for overall evaluation.

The collected feedback from this survey will be used to identify and improve on the weaker parts of the framework, making the system more usable. It will also allow us to evaluate how well understood are the most important concepts, so that we may improve the documentation and training material.

This event was a success, allowing us to collect very useful feedback. The following preliminary conclusions are highlighted:

* Participants found Hyperties are easily integrated into Apps. However, more examples are required
* All participants considered reTHINK easy to use
* The reTHINK library logging is too verbose
* Documentation needs to be improvement
* All participants expressed the intention to use the reTHINK framework again in future projects
* It would be great if reTHINK was supported in more browsers
* The authentication GUI needs improvements in terms of user interface
* The Hyperty toolkit was appreciated by participants
* It would be great if reTHINK had built-in integration with front-end frameworks and examples

### Survey - Teams presentation and background

This section of the survey was used to collect information about the participants. Participants could enter the competition alone or in teams of two elements. In total, this event had twelve participants divided into eight teams. Not all of them answered all the questions.

1. How old are you?

* 20-25 (8)
* 25-30 (1)
* >70 (1)

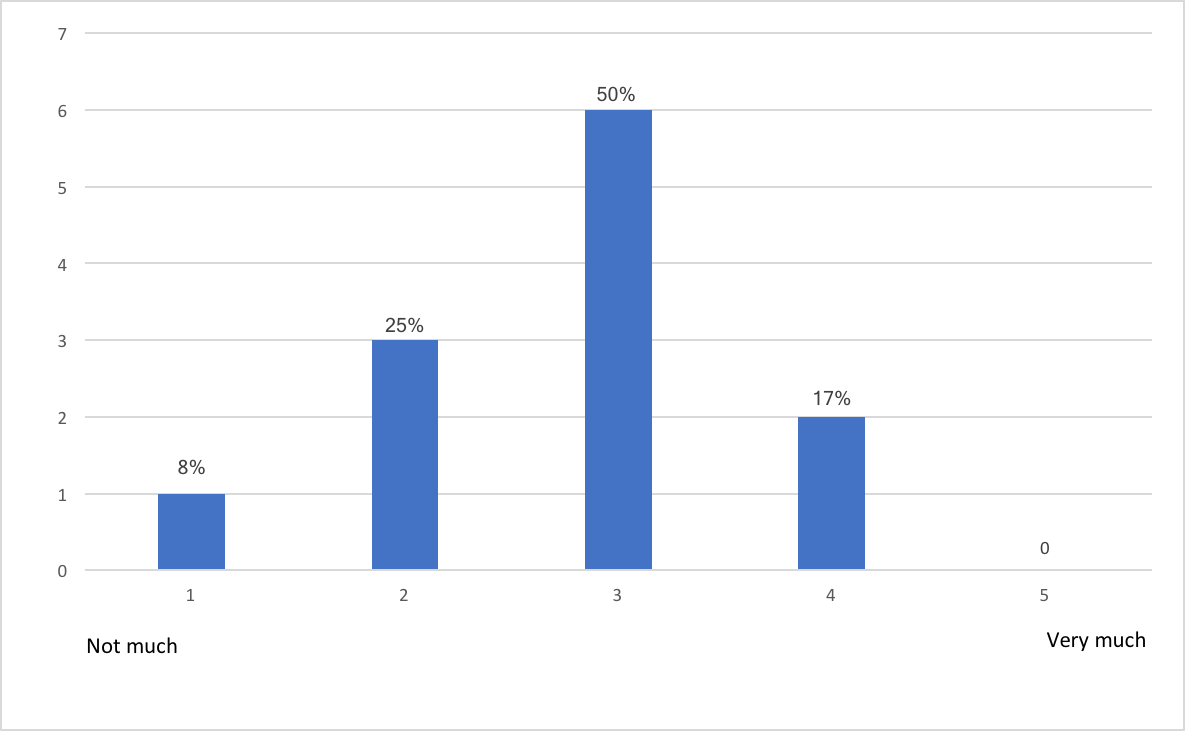
1. What is your genre?

* Male (11) - 92%
* Female (1) - 8%

1. What is your academic background?

* Electronics and Electrical Engineering (2) - 18%
* Computer Science Engineer (7) - 64%
* Telecommunications Engineer - 9%
* ENGINEER ( X67 + ENPC72 ) - 9%

1. How much knowledge do you have about web application development?

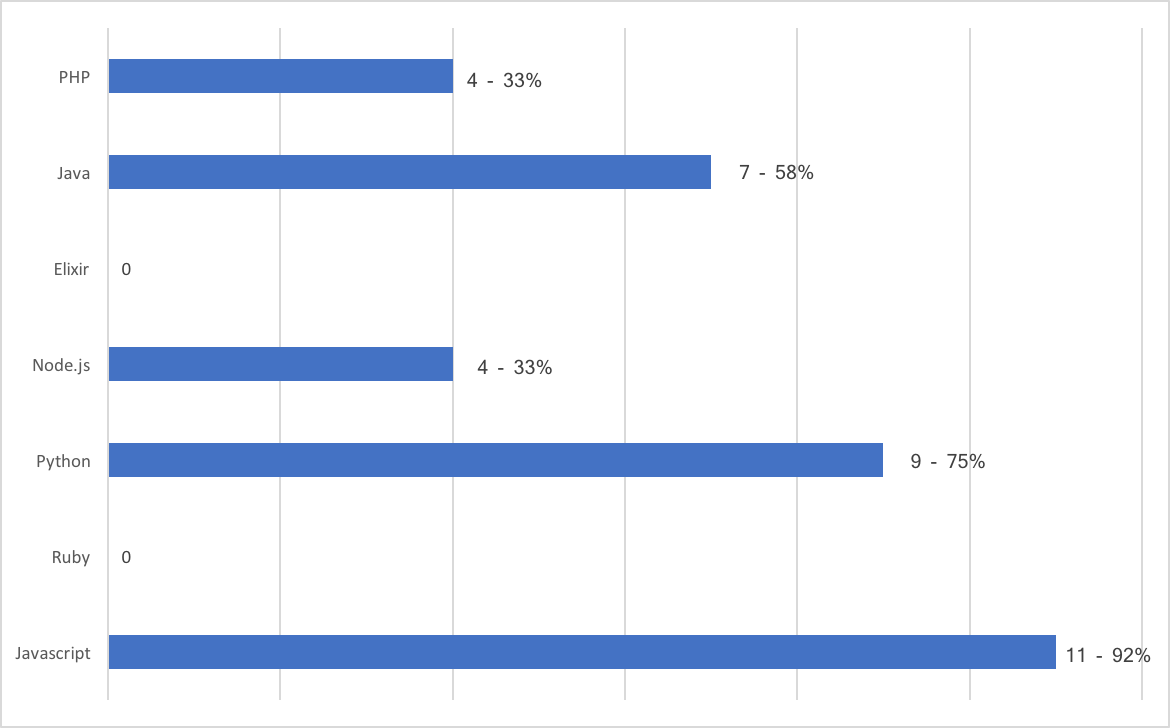


1-5

1. Do you have any experience with frameworks for web application development?

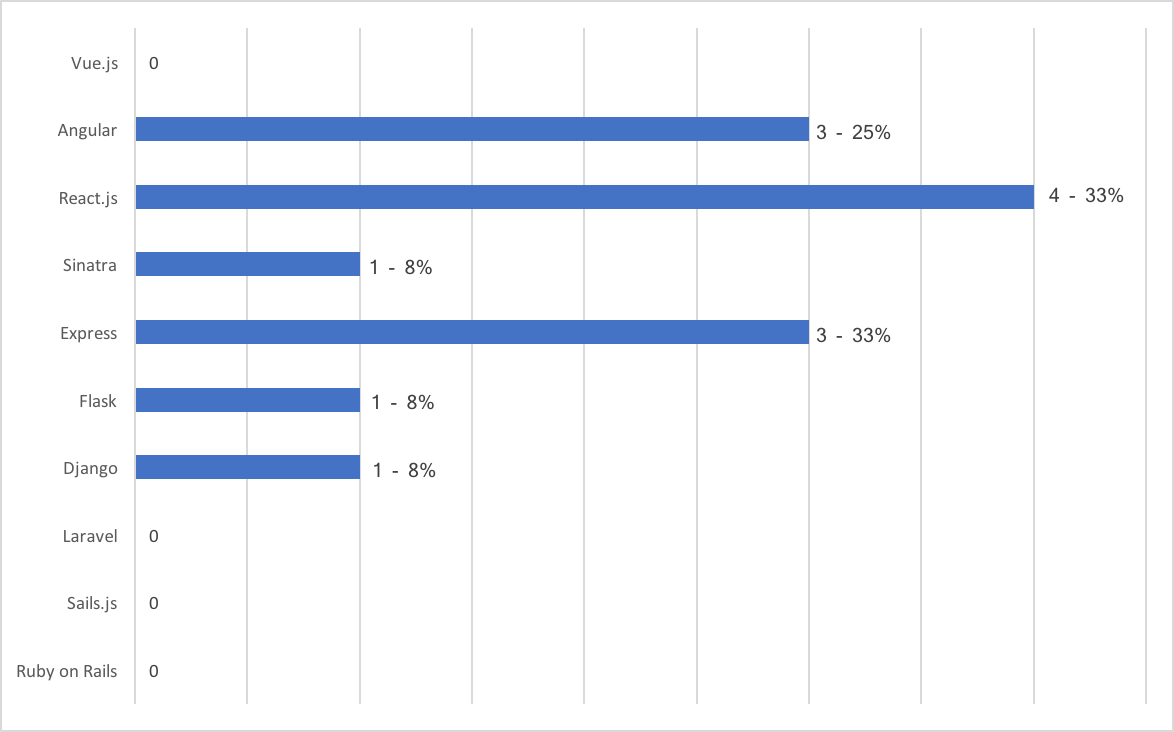
* Yes (5) - 58%
* No (7) - 42%

1. Select all the programming languages that you already work with.



1-7

1. Select all the frameworks for web application development that you already work with.

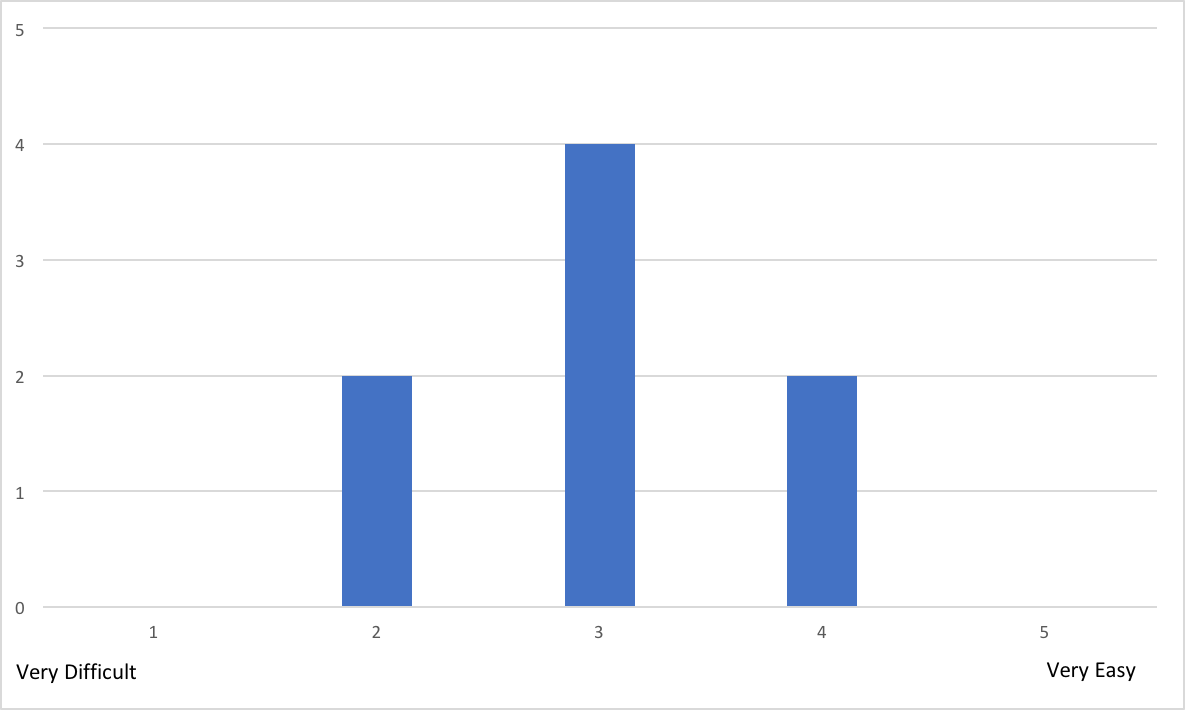


1-8

### Survey - First challenge

This section corresponds to the first challenge of the event that each team had to complete. The main goal of this first challenge was to provide a first contact with the reTHINK framework. Each team had to developed a simple web application that uses an hyperty called CodeGeneratorReporter, taking advantage of its features. This hyperty provides a numerical code depending of the team's name, that should be passed as input. This challenge was divided into two tasks. The first task was to install the reTHINK framework and to load the CodeGeneratorReporter hyperty. The second task was to use the features of the loaded hyperty.

1. How easy was to include Hyperties in your application?

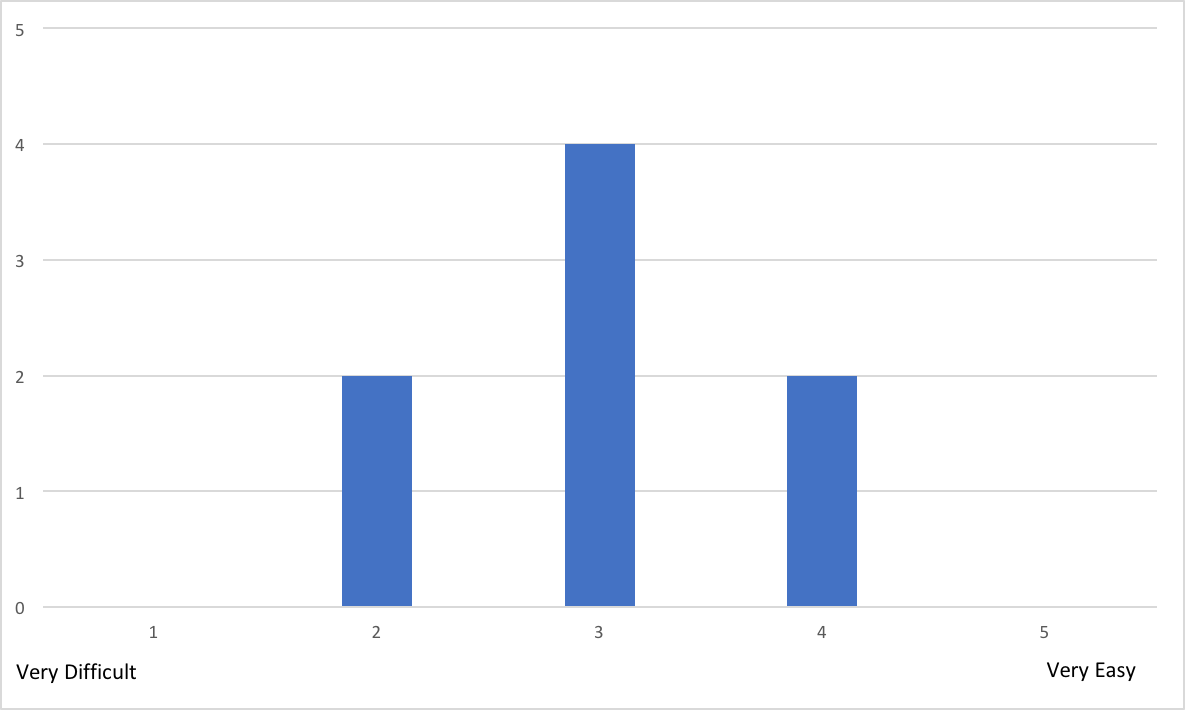


2-1

Most relevant comments to the previous question:

* Without the tutorial and further instructions, we would have taken a lot more time.
* After launching and installing, it was difficult to understand that authentication was needed and where to find a button to authenticate using Google

1. How much knowledge about reTHINK do you think is required to use it?



2-2

Most relevant comments to the previous question:

* You have to have minimal understanding of the framework (conceptually) to be able to understand how to use it.
* It's needed a good knowledge because is lacking some documentation

1. How much time did you spent reading the documentation? (estimate)

* 10min
* 15min (2)
* 20min (2)
* 30min
* 1h
* 2h

1. Did you manage to easily test and debug your application?

* Yes - 100%
* No - 0%

Most relevant comments to the previous question:

* The application itself was simple enough that testing and debugging were not a big problem. However, the rethink library logging is maybe a little too verbose.

1. How much time did you spent in the 1st task? (estimate)

* 10min
* 15min (2)
* 20min
* 30min
* 50min
* 1h
* 3h

**Note: Our estimate for this task was 45 minutes. The average time of the participants was 48 minutes.**

1. How much time did you spent in the 2nd task? (estimate)

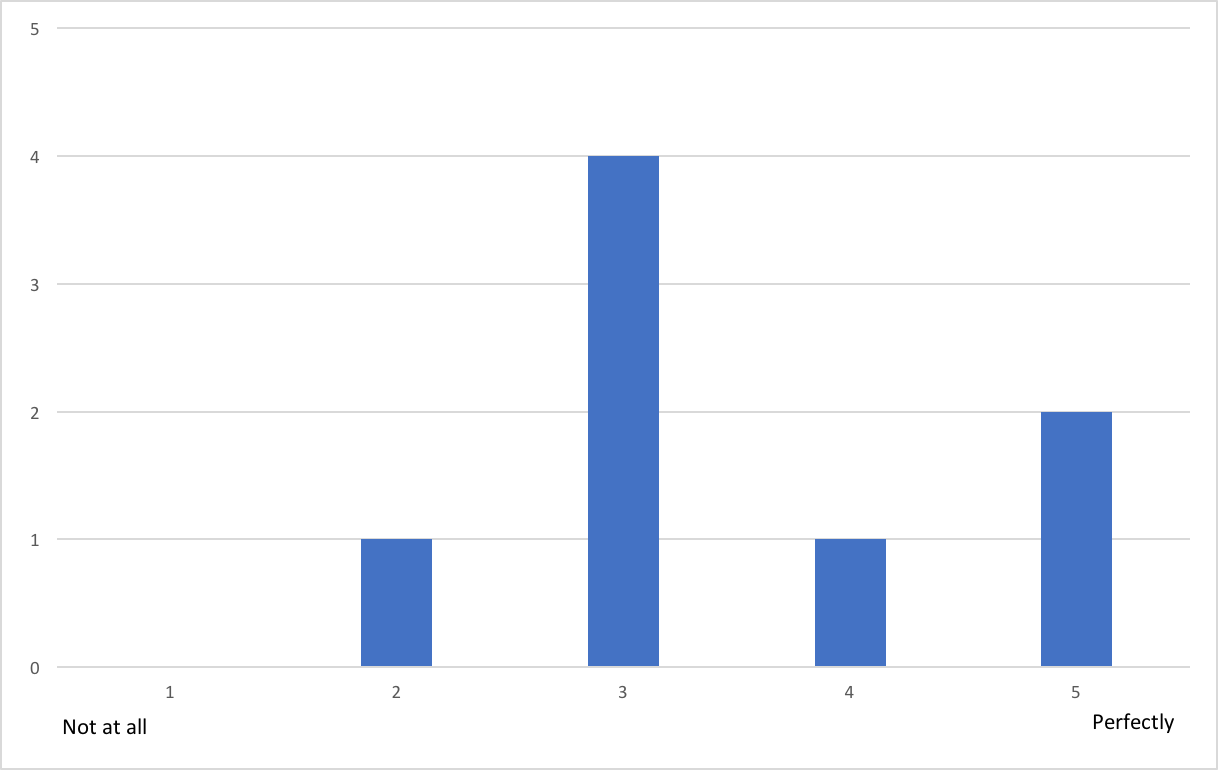
* 5min
* 10min
* 20min
* 35min
* 1h
* 1h45min
* 2h (2)

**Note: Our estimate for this task was 30 minutes. The average time of the participants was 44 minutes.**

1. Which task gave you the most trouble? And why?

* No problem (4)
* Task 1, due to misunderstanding of the script (2)
* The second task was harder because of some misunderstandings (2)

1. Did you manage to use the Identity Management features?



2-8

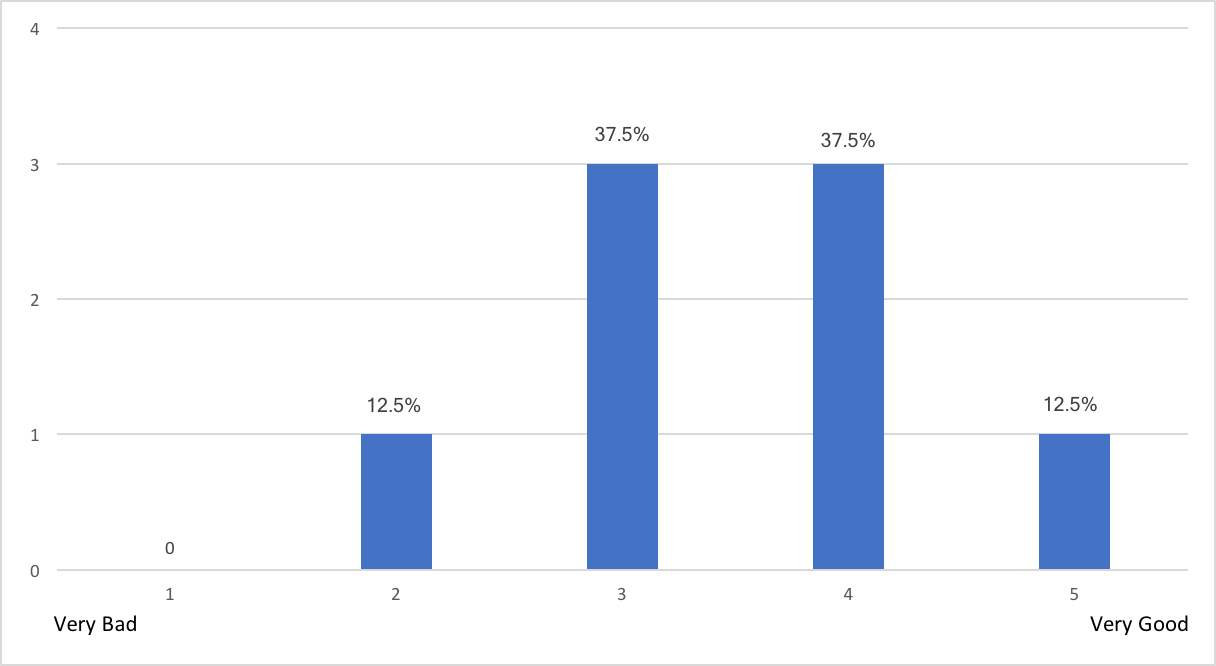
Most relevant comments to the previous question:

* Unfortunately only google appeared to work.
* It was difficult to understand that authentication was needed and where to find to google button to authenticate
* It did not seem to work very well. For example, I tried to login using Slack, but it would always open a Google authentication page, even though the page clearly displayed Slack as the IDP. It was also not possible to add a second Google account.

1. Let us know what limitations you found and how you would improve it.

* It would be useful (at least for developers) to be able to use anonymous identities.
* Excessive Logging
* When a user can authenticate with the Google Services, a dedicated button should be displayed, instead of being inside a dropdown menu
* Better documentation and some good tutorials to get people started
* The main limitation I identified was with the identity management feature

1. Classify your experience with reTHINK in this first challenge.



2-10

#### Conclusion:

Each team completed with success this first challenge and we were able to collect very useful feedback about reTHINK and its features. There were positive aspects, such as:

* Even with minimal knowledge about web application development, all participants were able to use reTHINK and complete this challenge;
* All participants considered reTHINK easy to use;
* All participants considered it easy to load Hyperties into their web applications.

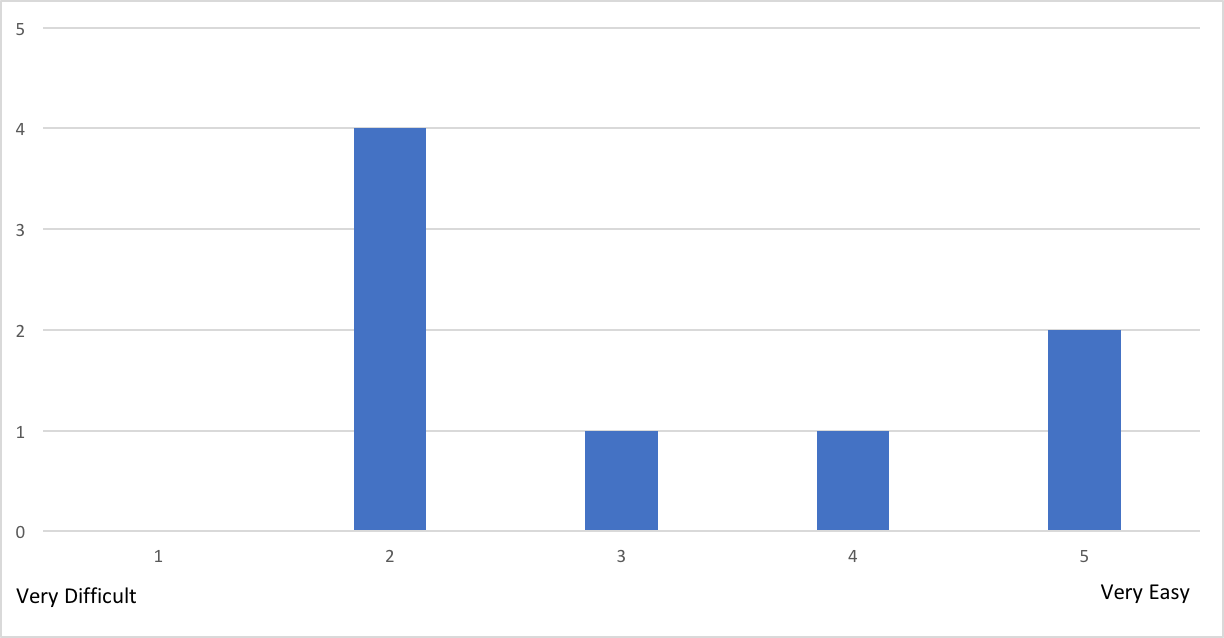
However, we also collected some negative feedback that should be addressed, such as:

* The authentication GUI, in terms of user interface, was difficult to understand for some participants;
* Without the tutorials and instructions available during the event, the participants recognized that it would have taken them much longer to complete all the tasks;
* Lack of documentation;
* The reTHINK library logging is maybe a little too verbose.

### Survey - Second challenge

This section corresponds to the second challenge of the event that each team had to complete. The main goal of this second challenge was to combine multiple hyperties into a web application. Each team had to developed a more complex web application, in terms of user interface, that uses the CodeGeneratorReporter and GroupChatManager hyperties. This challenge was divided into three tasks. The first task was to load two hyperties. The second task was to develop all the user interface to deal with all the features of the GroupChatManager hyperty e.g. creation of chat rooms, sending messages (chat capabilities). And finally, the third task was to combine the two loaded hyperties.

1. How easy was it to include more than one Hyperty in your application?



3-1

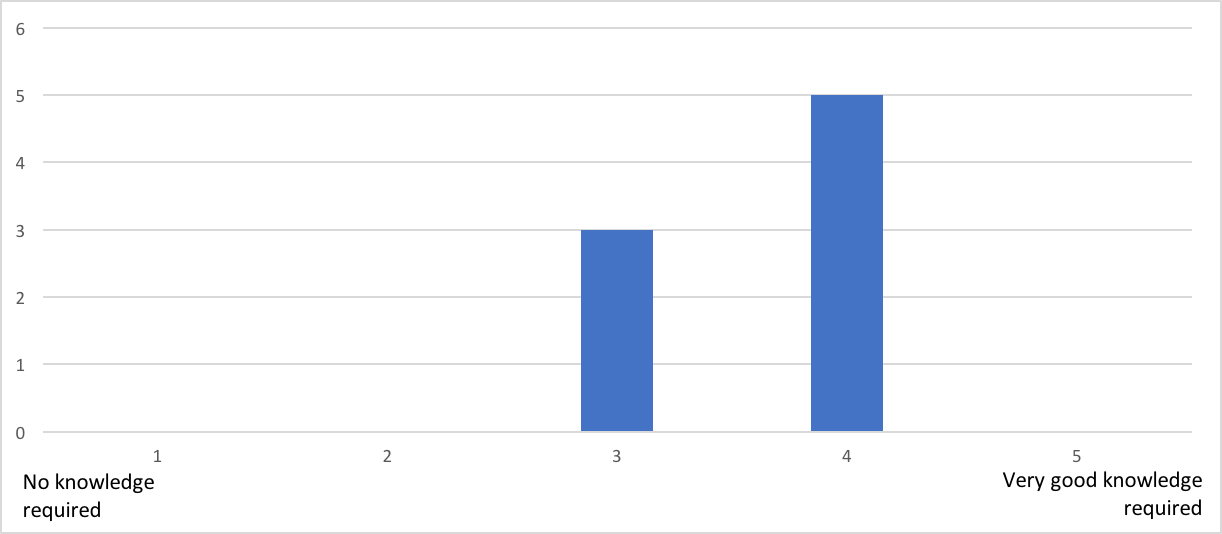
1. Were the used Hyperties well documented?

* Yes - 62,5%
* No - 37,5%

Most relevant comments to the previous question:

* The documentation is OK, but sometimes not entirely clear.

1. How much knowledge about reTHINK do you think is required to use it?



3-3

Most relevant comments to the previous question:

* You can get it working with only basic knowledge, but then you may find some things a bit confusing (for example, the domain when inviting users).

1. How much time did you spend reading the documentation? (estimate)

* 5min (2)
* 15min
* 20min
* 30min
* 1h
* 2h
* 3h

1. How much time did you spent in the 1st task? (estimate)

* 5min (2)
* 10min (2)
* 20min (2)
* 30min
* 1h

**Note: Our estimate for this task was 15 minutes. The average time of the participants was 19 minutes.**

1. How much time did you spent in the 2nd task? (estimate)

* 30min
* 1h
* 1h40min
* 2h
* 2h15min
* 3h (2)

**Note: Our estimate for this task was 150 minutes. The average time of the participants was 115 minutes.**

1. How much time did you spent in the 3rd task? (estimate)

* 5min (2)
* 30min (2)
* 40min
* 45min

**Note: Our estimate for this task was 15 minutes. The average time of the participants was 26 minutes.**

1. Which task gave you the most trouble? And why?

* The second task, especially the user join (2)
* In the third task there was a small problem when trying to authenticate with an anonymous mozilla tab were the menu would not normally appear, giving an error on the console, in chrome it worked as normal (2)

1. If you had to implement another chat application would you choose the reTHINK framework again?

* Yes - 100%
* No - 0%

1. Justify your answer to the previous question.

* With the experience we have now, it is quite easy and intuitive to implement this functionality using reTHINK.
* Ease of use
* Even though it was hard to start, with some practice from myself the framework would be great to make new projects in the future
* After realizing the operation is relatively simple and simplifying the use compared to the traditional method
* It is relatively simple to use but needs better documentation
* The yes is actually a maybe. I think there are some good things in reTHINK. It was very easy to create a chat room, invite other users and send messages. However, I am concerned about the stability of the platform. I often experienced timeouts and strange errors that would only go away after clearing the cache.

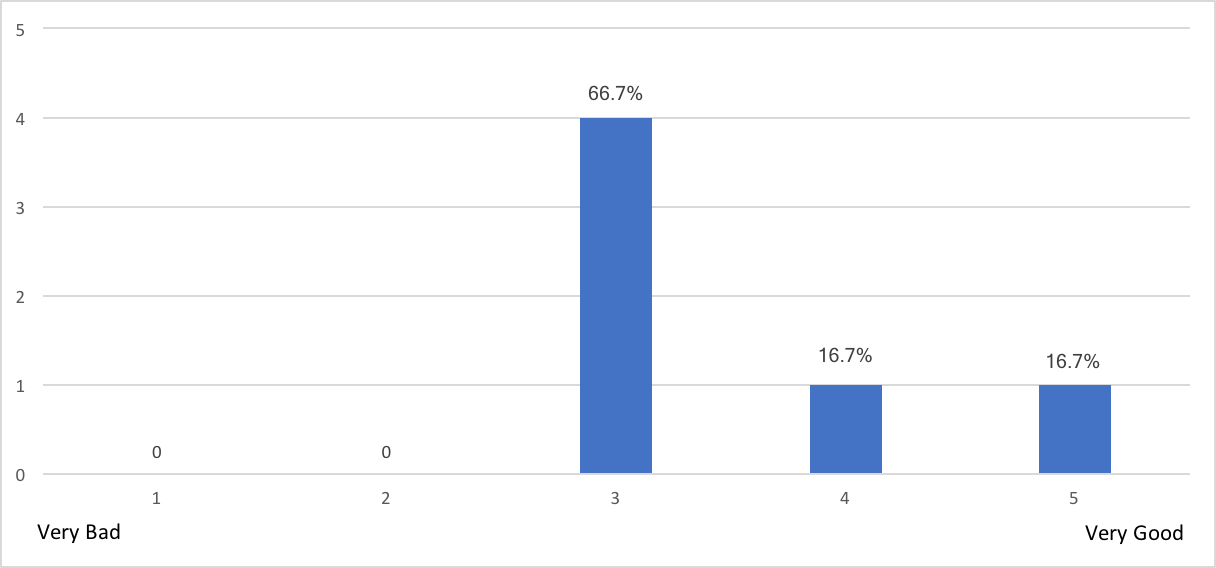
1. If your answer was "No" to question 9, which framework/libraries would you choose then? And why?

* I'm not sure. I'm not very familiar with the options out there. I might take a look at XMPP, but I would probably not want to use it for a simple chat application on the web.

1. Let us know what limitations you found and how you would improve it.

* Async system not fully functional
* The only problem I had with the framework was when trying to authenticate with the anonymous firework page
* It's mostly lack of documentation and having to use ".instance" to access methods is unintuitive
* Frequent errors/timeouts.

1. Classify your experience with reTHINK in this second challenge.



3-13

#### Conclusion:

Only four teams were able to complete this second challenge with success. In spite of this, it was still possible to collect some useful feedback about reTHINK and its features. There were positive aspects, such as:

* The participants recognized that if they had a bit more practice with the framework, they would use it in future projects;
* All participants would use reTHINK framework again to develop new chat applications;
* It is quite easy and intuitive to implement chat functionality using reTHINK.

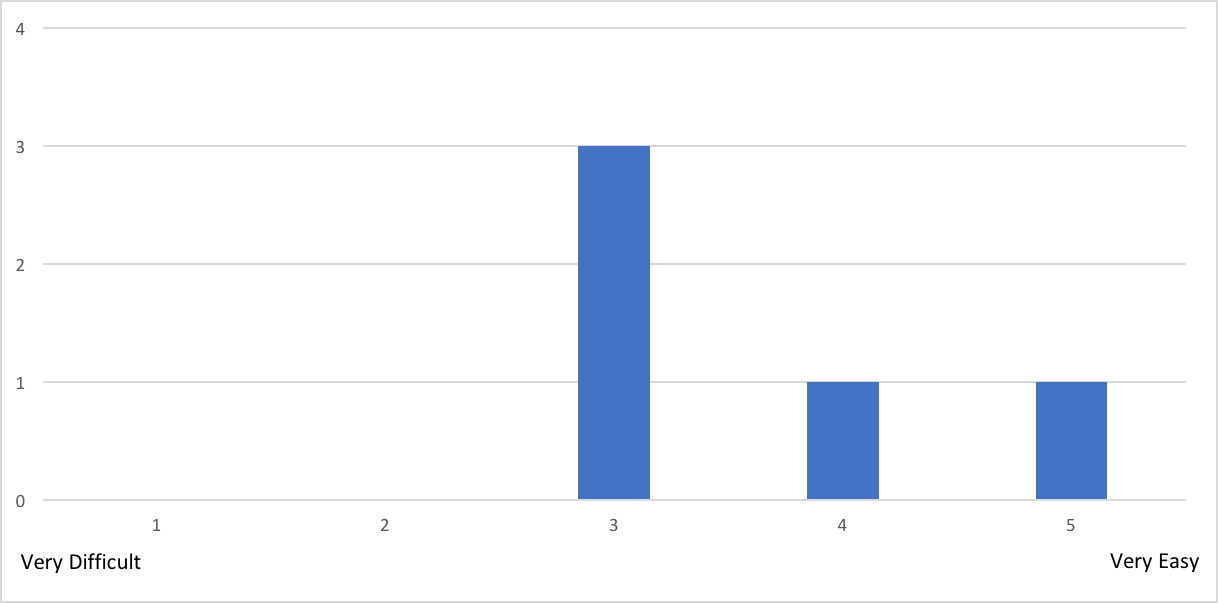
However, we collect also some negative aspects that should be considered, such as:

* Lack of documentation and sometimes the existing documentation is not entirely clear;
* Problems arise if the used browser is not Chrome;
* Some timeout errors and no explanation for them;
* The fact that it is necessary to clean the cache constantly is a bit dull.

### Survey - Third challenge

This section corresponds to the third challenge of the event that each team had to complete. The main goal of this third challenge was to change the CodeGeneratorReporter hyperty functionality, reusing the previous developed application. Now, this hyperty should generate a code and a timestamp according to the team's name, that should be passed as input. This challenge was divided into two tasks. The first task was to configure the hyperty toolkit, following the available documentation on the reTHINK repository. The second task was to change the hyperty functionality and to use it in the developed application.

1. How easy was it to setup the reTHINK environment in your local machine?

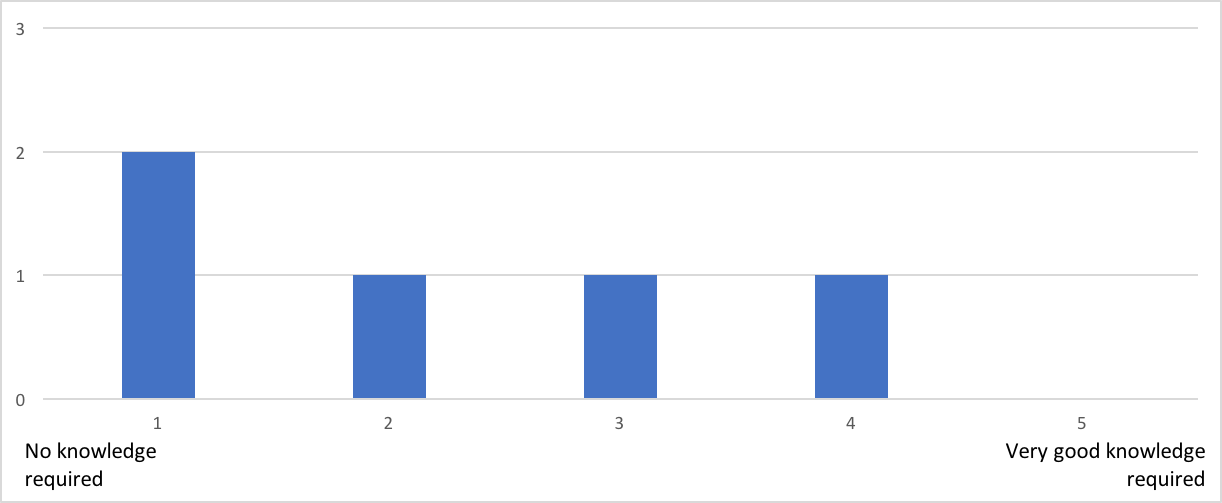


4-1

Most relevant comments to the previous question:

* We had problems installing the correct branch (so, our fault), but the branch had a mistake.

1. How much knowledge about reTHINK did you think is required?



4-2

1. How much time did you spent reading the documentation? (estimate)

* 10min (2)
* 30min
* 3h

1. Did you have problems with the reTHINK environment configuration? Justify.

* No (2)
* No, most problems were small mistakes on the developer's end.
* My Ubuntu 17.04 did not have a "node" binary, instead it is called "nodejs". I had to apt install nodejs-legacy for the build to work. I also had a problem when running it because it was trying to bind to a low port (<1024). I figured it out pretty quickly, but the error message is not very descriptive. I also eventually found a note about this in the Wiki, but by that time I had already Googled the problem and solved it. It would be better to mention it in the README, because everyone who uses Linux is going to have the same problem.

1. How much time did you spent in the 1st task? (estimation)

* 25min
* 30min
* 1h
* 1h30min

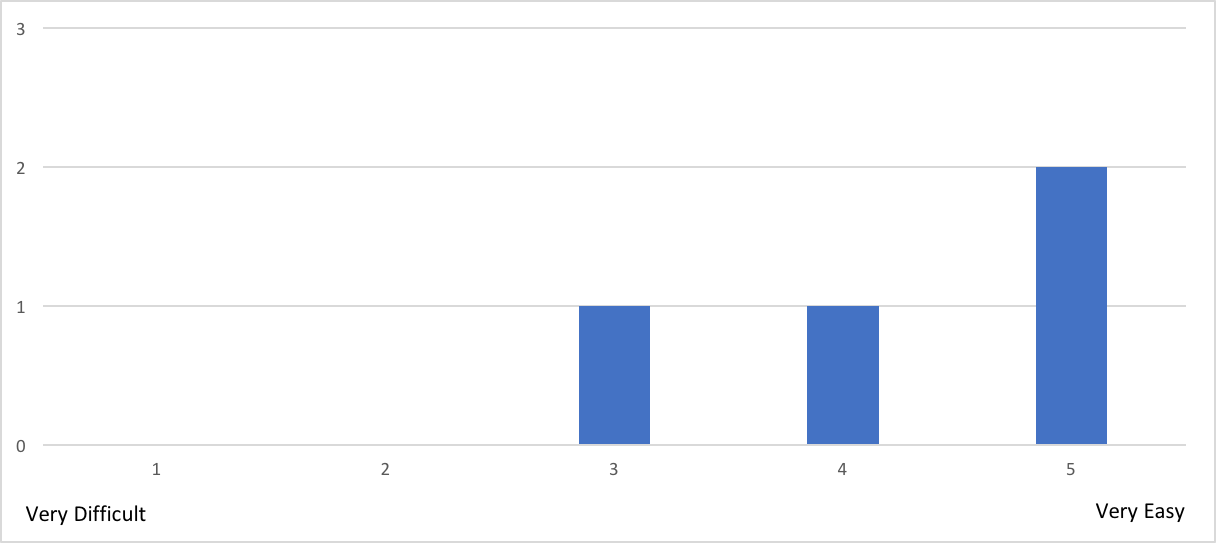
**Note: Our estimate for this task was 35 minutes. The average time of the participants was 51 minutes.**

1. How much time did you spent in the 2nd task? (estimation)

* 0min
* 5min
* 20min
* 3h

**Note: Our estimate for this task was 40 minutes. The average time of the participants was 51 minutes.**

1. How easy was it to change the Hyperty functionality?

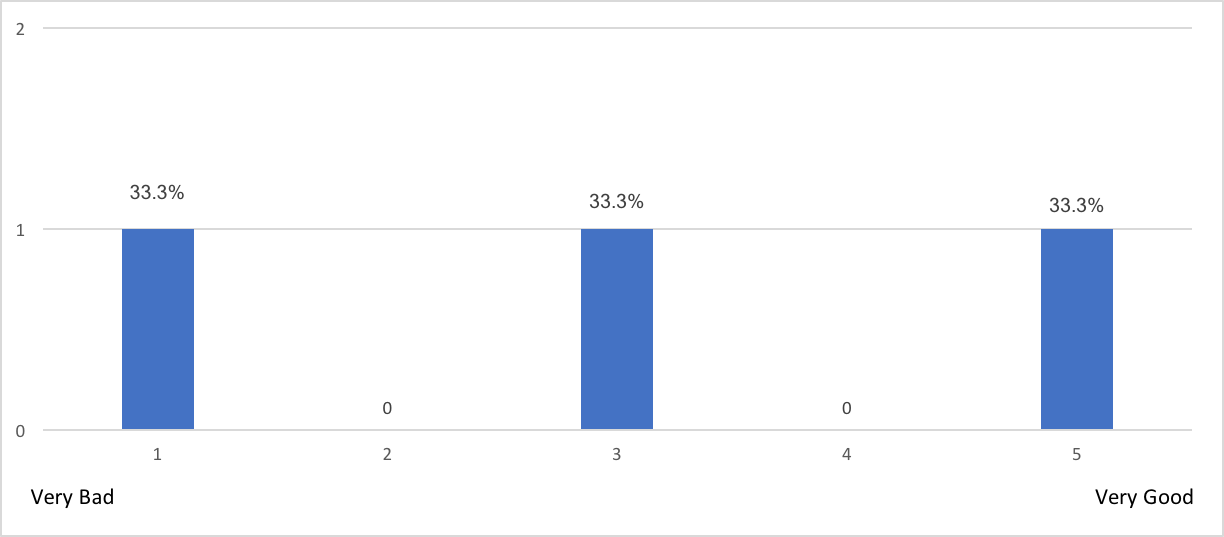


4-7

1. Let us know what limitations you found and how you would improve it.

* Only being able to have 1 tab open loading a certain hyperty lost us a bit of time.

1. Classify your experience with reTHINK in this third challenge.



4-9

#### Conclusion:

Only two teams were able to complete this third challenge with success. In spite of this, it was still possible to collect some useful feedback about the development of hyperties. There were positive aspects, such as:

* Most of the participants considered it easy to setup the reTHINK development environment (hyperty toolkit);
* Most of the participants considered it easy to change the hyperty functionality.

However, we collect also some negative aspects that should be considered, such as:

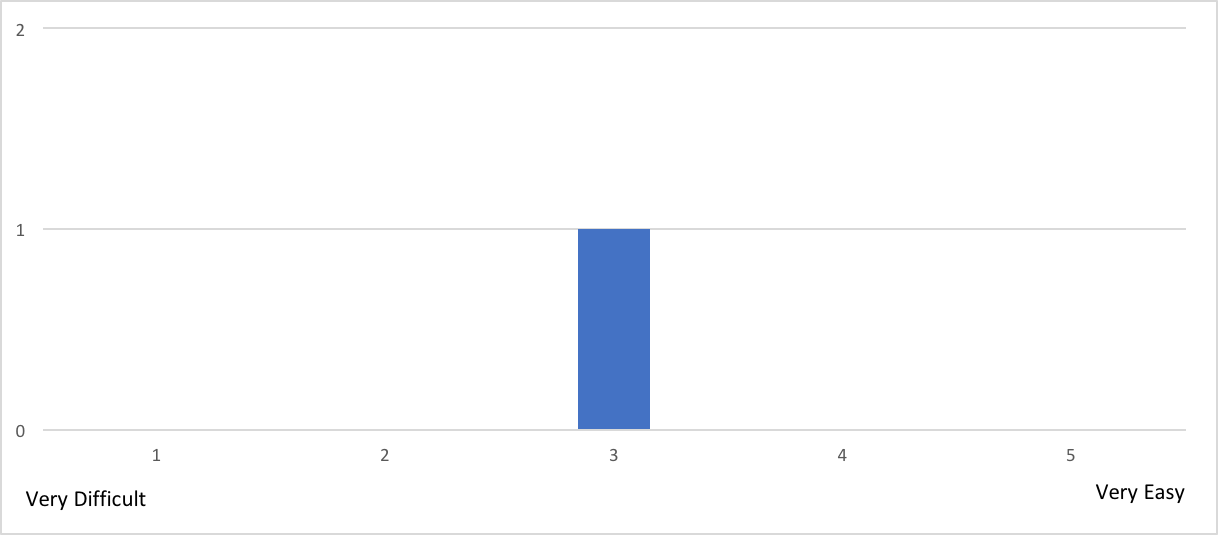
* Lack of documentation;
* Only being able to have 1 tab open loading a certain hyperty.

### Survey - Fourth challenge

This section corresponds to the fourth challenge of the event that each team had to complete. The main goal of this fourth challenge was to develop a video conference application. This application should cover some functional requirements, such as the creation of chat rooms with the possibility of inviting users to join in, A/V calls between users and the possibility to make A/V calls at the same time that a chat room is available (exchanging messages). This was an open challenge in which the participants could choose hyperties already available or develop their own hyperties.

Unfortunately, neither of the teams that made it to this challenged, managed to finish this challenge in time. Below are listed all the questions and some of the collected answers.

1. Did you choose to develop new Hyperties? If yes, describe them briefly. If not, pass to the sixth question.
2. How easy was to develop new Hyperties?
3. How much knowledge about reTHINK did you think is required?
4. How much time did you spent reading the documentation? (estimate)
5. Which features did you tried to implement that gave you the most trouble? After answer to this question, pass to the tenth question.
6. Which Hyperties did you use?
   * CodeGeneratorReporter, GroupChatManager and Connector
   * GroupChatManager
7. In your opinion, how hard would it be to develop new hyperties?



5-7

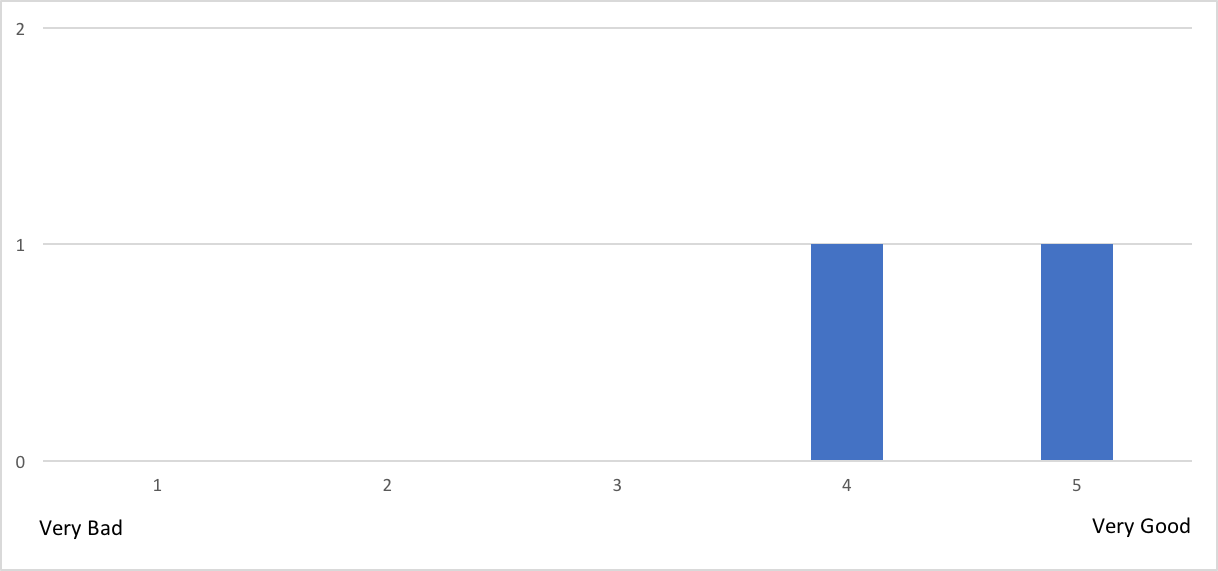
Most relevant comments to the previous question:

* We didn't have time to attempt to create a hyperty from scratch, so we cannot infer the difficulty. Based on changes made in challenge 3, it seems manageable
* It's hard to tell. Modifying the hyperty in challenge 3 was very easy, but I did not have the chance to build a new one from scratch. Still, looking at the code of the existing hyperties, it does not seem too difficult.

1. Did you try to change or add new functionalities to the used Hyperties?

* No (2)
* Yes (0)

1. If your answer to the previous question was "yes", what did you try to implement?
2. Let us know what limitations you found and how you would improve it.
3. Classify your experience with reTHINK in this fourth challenge.



5-11

### Survey - Fifth challenge

This section corresponds to the fifth challenge of the event that each team had to complete. The main goal of this fifth challenge was to develop a new hyperty from scratch. Each team was free to design the hyperty, as well as the web application that uses it.

Unfortunately, no team reached this challenge, so we could not collected answers. Below are listed all the questions.

1. Describe briefly the developed web application.
2. Describe briefly the developed hyperties.
3. How easy was to develop new Hyperties?
4. How much knowledge about reTHINK did you think is required?
5. How much time did you spent reading the documentation? (estimate)
6. How useful and complete is the available documentation to support the development of Hyperties?
7. Please provide your suggestions to improve the documentation.
8. Which features did you tried to implement that gave you the most trouble?
9. Classify your experience with reTHINK in this fifth challenge.

### Survey - reTHINK Overall Evaluation

This section of the survey was used to make an overall evaluation about reTHINK.

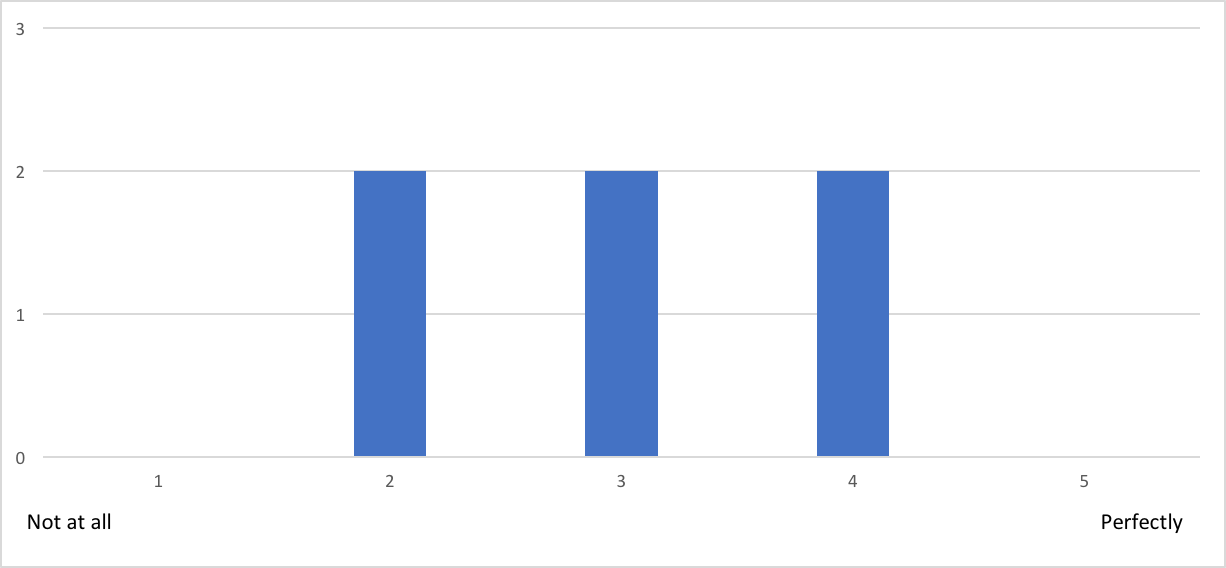
1. Which framework for web application development would you recommend? And why?

* reTHINK
* Never used frameworks to develop web applications
* We haven't used other web frameworks
* ReactJs, as it has a healthy community, it is well supported, large heaps of documentation and the developer experience in general is very pleasant.
* I don't have enough experience with web frameworks to make a recommendation

1. Taking into account your experience in web application development, how do you compare reTHINK framework with other web frameworks?

* I liked reTHINK but hadn't much experience with others
* We haven't used other web frameworks
* All the challenges we did were easy to implement with reTHINK. Unfortunately, we couldn't experience how it would work when used with a real-life application, but we consider it would be manageable. It needs more extensive documentation, especially on the demos.
* Again, not enough experience to compare.

1. How easy was it to understand the reTHINK documentation?



6-3

1. What features of the reTHINK framework look more promising to you?

* The fast chatrooms creations
* It's ease to use in projects. Doesn't require much to run in a script
* We think the most important feature (killer-app) of the reTHINK framework is the possibility of completely different and independent developers developing compatible applications without needing to standardize protocols
* The best thing about reTHINK is how simple it is to take the "plug-and-play" hyperties and build an application out of them

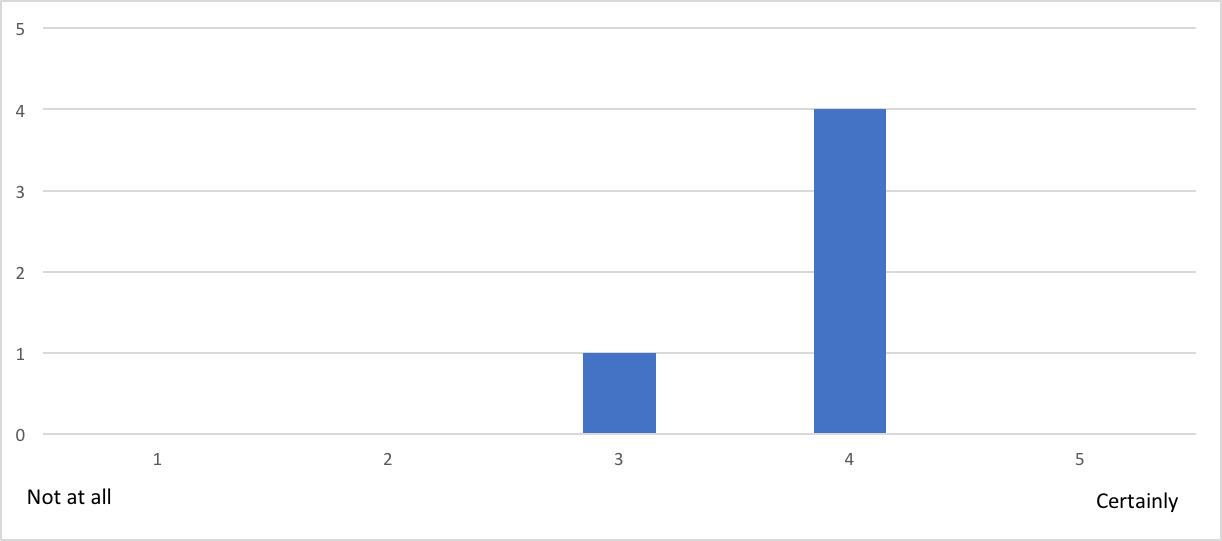
1. In your opinion, what are the main benefits of using Hyperties?

* The mainly benefits is maybe the with some training you could achieves good results fast
* It's easy to use Hyperties in an application
* Hyperties shine when developing complex applications that need to interact with different domains. It simplifies the process of managing connections, domains, etc, which greatly improves the developer experience
* Easy discovery. Things seem to "just work" (at least when they don't timeout)

1. What features would you like to see in reTHINK framework?

* For now none, but we still have to explore the framework better
* Built-in integration with front-end frameworks and examples
* More hyperties! I think the best way to popularize it would be to have the basic building blocks ready, so you can quickly build an application out of them

1. Would you recommend the reTHINK framework?



6-7

#### Conclusion:

Taking into account all the feedback, the overall evaluation of reTHINK seems promising. There were positive aspects, such as:

* Most of the participants were very enthusiastic about the framework;
* Most of the participants suggested new features.

However, we collect also some negative aspects that should be considered, such as:

* The documentation needs a few improvements in order to be more understandable;
* Participants would like to see more hyperties in reTHINK.