



Free version: Low quality pictures

vnpvproject.wordpress.com

Contents

1	2016	5
1.1	October	6
	Blog (2016-10-07 10:57)	7
	First Blogentry (2016-10-07 10:57)	8
	Contact (2016-10-07 10:57)	9
	About VNV (2016-10-07 10:57)	10
	Project vision of VNV (2016-10-11 12:23)	11
	Project Roles (2016-10-12 11:42)	12
	Software Requirements Specification (2016-10-24 08:41)	14
	Use Cases (2016-10-31 13:37)	16
1.2	November	17
	Scrum (2016-11-03 08:32)	18
	Cucumber (2016-11-14 09:01)	19
	Class Diagram (2016-11-21 10:58)	21
	MVC Architecture (2016-11-26 23:11)	22
1.3	December	23
	Gantt Chart (2016-12-04 15:52)	24
	Midterm Handin (2016-12-11 15:45)	25
2	2017	29
2.1	April	30
	We are getting started again (2017-04-17 19:09)	31
2.2	May	33
	Function points (2017-05-01 18:38)	34
	Test plan (2017-05-07 18:10)	36
	Refactoring (2017-05-15 10:12)	37
	Pattern (2017-05-22 16:36)	38
	Metrics (2017-05-29 16:54)	40
2.3	June	43
	Test Coverage (2017-06-12 12:39)	44
	Continuous Integration (2017-06-12 14:41)	45
	Final (2017-06-19 09:43)	46

1. 2016

1.1 October

Blog (2016-10-07 10:57)

Auf dieser Seite finden Benutzer den Blog deiner Website.

Dies ist nur ein kurzer Auszug deines Blogs.

First Blogentry (2016-10-07 10:57)

This blog is now online

Contact (2016-10-07 10:57)

If you have any comments or suggestions please contact us.

```
[contact-form][contact-field label="Name" type="name" required="1" /][contact-field label="E-Mail-Address" type="email" required="1" /][contact-field label="Website" type="url" /][contact-field label="Comment" type="textarea" required="1" /][contact-form]
```

Dies ist nur eine Kurzfassung der Seite „Kontakt“.



We, Patrick, Waldemar and Gabriel, are students at the Baden-Württemberg Cooperative State University Karlsruhe and develop VNV as part of the course Software Engineering.

VNV is an acronym for Verleihen, Dienstleistungen, Verschenken (lending, services, comping). VNV will be a web portal, where you can connect with your friends to borrow items from them, give things you don't use anymore to a friend looking for exactly that one or you can check if one of your mates can help you with a problem (e.g. if you need help with your network).

Dies ist nur eine Kurzfassung der Seite & „Über“.

Project vision of VNV (2016-10-11 12:23)

We, Patrick, Waldemar and Gabriel, are students at the Baden-Württemberg Cooperative State University Karlsruhe and develop VNV as part of the course Software Engineering.

VNV is an acronym for Verleihen, Dienstleistungen, Verschenken (lending, services, comping). VNV will be a web portal, where you can connect with your friends to borrow items from them, give things you don't use anymore to a friend looking for exactly that one or you can check if one of your mates can help you with a problem (e.g. if you need help with your network).

Therefore we have to implement functions for registration, account activation, login, adding friends, adding new entries with an offer or request, communication with your friends, ...

We will develop the server application in Java with the Spring Framework.

Possible future features could be:

- connection to Google/Facebook
- calendar features
- a mobile app

Jan (2016-10-12 10:45:46)

Good idea with a clear objective! Also, you give a good overview of your project vision and the various subparts.

Andreas (2016-10-12 11:04:41)

Hi VNV Team, I like your idea and your blog looks great too. I was able to get a first impression of your project and what your topic is! Looking forward to hear more from you guys and I will borrow some "Klempnerskills" for sure after release! Regards, Andreas

vnvproject (2016-10-12 11:16:15)

Hi Andreas, thanks for your feedback and the interest in our project. Cheers Gabriel from VNV

Midterm Handin – VNV-Project (2016-12-11 15:45:53)

[...] Project Vision (blog post) [...]

Final – VNV-Project (2017-06-19 09:43:26)

[...] Project Vision (blog post) [...]

Project Roles (2016-10-12 11:42)

In this post we will share the roles every team-member has and what technologies we'll use.

Roles:

- Project Manager: Waldemar
- Tool Specialist: Waldemar
- Deployment Manager: Gabriel, (Waldemar)
- Designer: Patrick
- Configuration Manager: Gabriel
- Implementation:
 - Main-Frontend-Developer: Patrick
 - Main-Backend-Developer: Gabriel
- Test Manager: Waldemar
- Tester: Waldemar, Patrick, Gabriel
- Blogposts: Gabriel
- Blog-Reviews: Waldemar, Patrick

Tools & Co:

- YouTrack
- GitHub
- Java, Spring
- IntelliJ
- Cucumber, Selenium, JUnit
- Redis, Neo4j
- AngularJS
- Bootstrap
- Maven
- Sonarqube
- TravisCI
- PackageCloud
- (Microsoft Project)

Andreas (2016-10-14 11:07:01)

Hi team, i just checked your roles and i think you have made good decisions. In my opinion all roles / tasks are covered and every of your member has a backup / partner to work with. Your choice of the tools you will work with seems also good. Maybe we can exchange some information / stuff about IntelliJ. Have a nice weekend and hope to hear more from you guys, Andreas

Tobias (2016-10-14 12:26:30)

Hey guys, your choices seem pretty much solid to me, but do you really think it's necessary to work with Change Management? We chose not to. Maybe we can compare experiences along the way. So long Tobias

BillChill (2016-10-26 10:44:04)

Hey guys, your roles seem to be well structured but be careful with the high amount of roles for each person. best regards

Midterm Handin – VNV-Project (2016-12-11 15:45:56)

[...] Project Roles (blog post) [...]

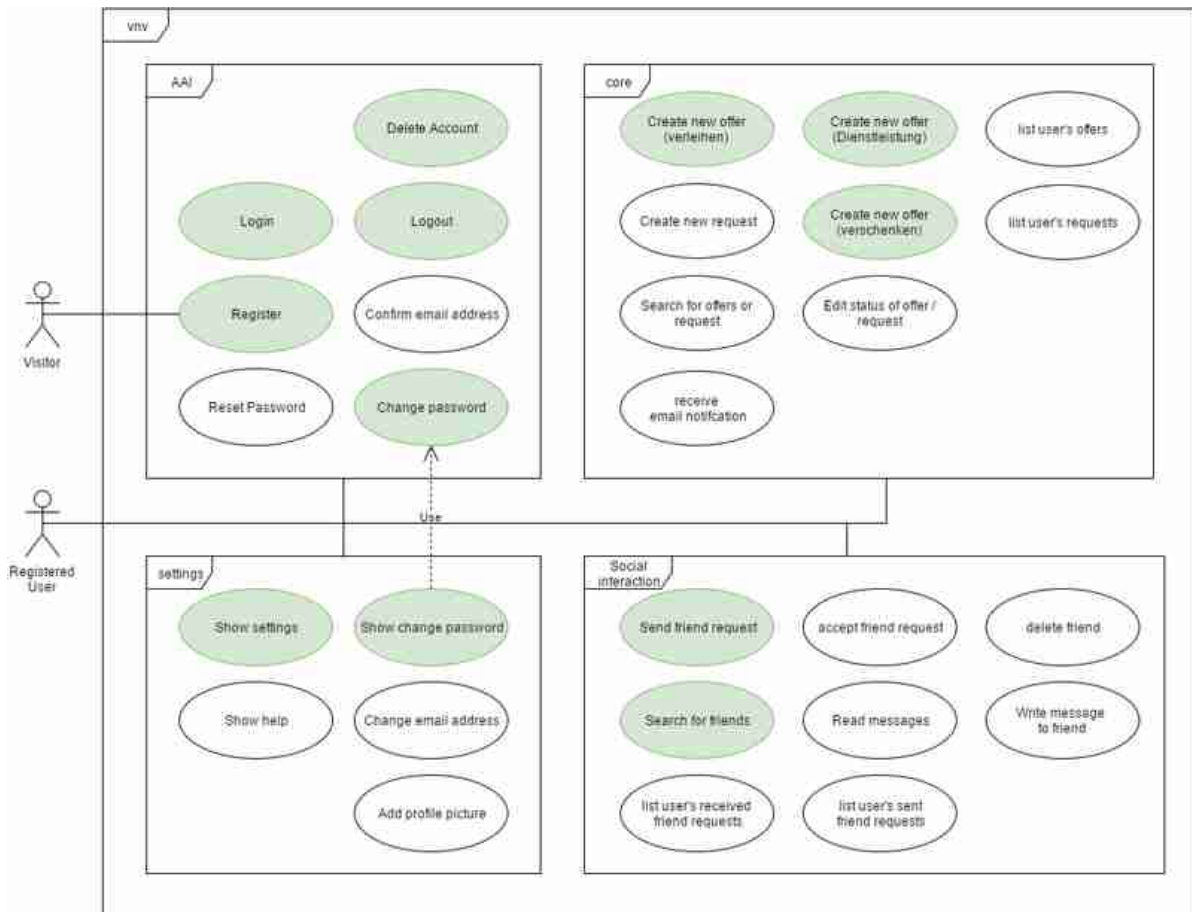
Final – VNV-Project (2017-06-19 09:43:27)

[...] Project Roles (blog post) [...]

Software Requirements Specification (2016-10-24 08:41)

This post is about the SRS of our project.

We want to show you our use case diagram:



You can find the newest version of this diagram [1]here and the detailed SRS [2]here.

Generally you can find all the documentation about our project on [3]Documentation-GitHub.

1. <https://github.com/WMerk/VnVProject/blob/master/doc/use%20cases/SRS.png>

2. <https://github.com/WMerk/VnVProject/blob/master/doc/SRS.MD>

3. <https://github.com/WMerk/VnVProject/tree/master/doc>

Tobias (2016-10-24 11:56:27)

Hi, From what I can tell your UC Diagram seems to cover all the necessary features. Your choice of optional UCs also lets you focus on the basic functions of vnv, so that you are most likely able to complete their implementation by the end of the year. I really liked your SRS Document. All the information was in there and it should easily be expanded if the need for changes arises during the project. It should serve you well! Though there are a few grammatical errors in there and your table of contents is not properly formatted. You may want to change that. Otherwise keep up the good work! Looking forward to hearing from you again.

vnvproject (2016-10-24 12:51:00)

Hi Tobias, thank you very much for your feedback. We incorporated your suggestion on the toc formatting. Cheers Gabriel

gtagroup (2016-10-24 13:00:04)

Hi, you did a good job with your UC diagram. It seems to be complete so you thought about everything. Maybe you could add an explanation which shows the color's meaning. Your SRS is also pretty good and clearly arranged. lovely greetings <3

joinsportsblog (2016-10-26 09:15:25)

Hi, I like the design of your UC diagram. It is clear structured and understandable for me. The SRS contains all relevant information, which is important for the actual process. For me it is nice how you described every single functionality of the UC diagram. Keep it up! Greetings, The JoinsSports-Team

Midterm Handin – VNV-Project (2016-12-11 15:46:00)

[...] Software Requirements Specification (blog post) [...]

Final – VNV-Project (2017-06-19 09:43:30)

[...] Software Requirements Specification (blog post) [...]

Use Cases (2016-10-31 13:37)

Hello everyone,

today we present you two of our use cases. They are also linked in the [1]SRS and we will link all future use cases there.

You can find the use case "Add a friend" [2]here.

And the use case "Create a new offer" [3]here.

1. <https://github.com/WMerk/VnVProject/blob/master/doc/SRS.MD>
2. https://github.com/WMerk/VnVProject/blob/master/doc/UC_AddFriend.MD
3. https://github.com/WMerk/VnVProject/blob/master/doc/UC_CreateNewOffer.MD

energycheetah (2016-11-01 13:56:31)

Hi, wow, I like your work. I don't find anything to criticize. It's easy to understand with the mock ups. Greetings Cheetah

Project Ortus (2016-11-01 19:54:38)

Hey VNV Team, We just checked your Use Cases and we all really liked it. The process of renting / borrowing something with your tool is very clean and nice visualized. The process flow is easy to follow and the parts of both users is clear. We also like the idea and the design of your mockups. Keep on and regards, Team ortus

Midterm Handin – VNV-Project (2016-12-11 15:46:04)

[...] Use Cases (blog post) [...]

Final – VNV-Project (2017-06-19 09:43:42)

[...] Use Cases (blog post) [...]

1.2 November

Scrum (2016-11-03 08:32)

Today we'd like to invite you to visit our [1]YouTrack. There you can watch our project management, see all the relevant issues and have a look on our [2]burndown this week.

We will have sprints of one week. Our aim is too have an working registration and login in two weeks. Hopefully registration will work at the end of the first sprint.

We decided to use YouTrack over Jira, because YouTrack has better integration with IntelliJ, our IDE. Furthermore our Tool Specialist already was familiar with YouTrack. So we were able to connect YouTrack with our [3]GitHub repo and with IntelliJ.

1. <https://vnm.myjetbrains.com/youtrack/issues>
2. <https://vnm.myjetbrains.com/youtrack/agiles/84-2/85-8?chart>
3. <https://github.com/WMerk/vnm>

energycheetah (2016-11-05 19:08:46)

Hi vnm team, I like your update on the tracking tool.. What I don't understand, on the dashboard is written 1 day left and 3 days left. What is the right one? And hasn't the sprint to be 2 weeks? Cheers, Cheetah

Waldemar (2016-11-07 08:38:28)

Hello enegycheetah, Thanks for your comment! The 3 left days are the sum of the estimation from the open issues which we wanted to close this sprint. The diagram shows a timeslot from monday to sunday. Thats where time conflict came from (our sprints are estimated from wednesday to thuesday). The size of the sprints can also be one week. We have a lot of work to do so we decided to set up our sprints as one-week-sprints. Regards vnm

Team Project Ortus (2016-11-07 16:17:19)

Hello VNV, we liked your post. You named the tasks very clearly. It is easy to see which tasks you are currently working on. We liked how you treat the working process. Especially your burndown diagramm looks very nice and organized. Keep it up. Regards, Team Project Ortus

Midterm Handin - VNV-Project (2016-12-11 15:46:07)

[...] Scrum (blog post) [...]

Final - VNV-Project (2017-06-19 09:43:43)

[...] Scrum (blog post) [...]

Cucumber (2016-11-14 09:01)

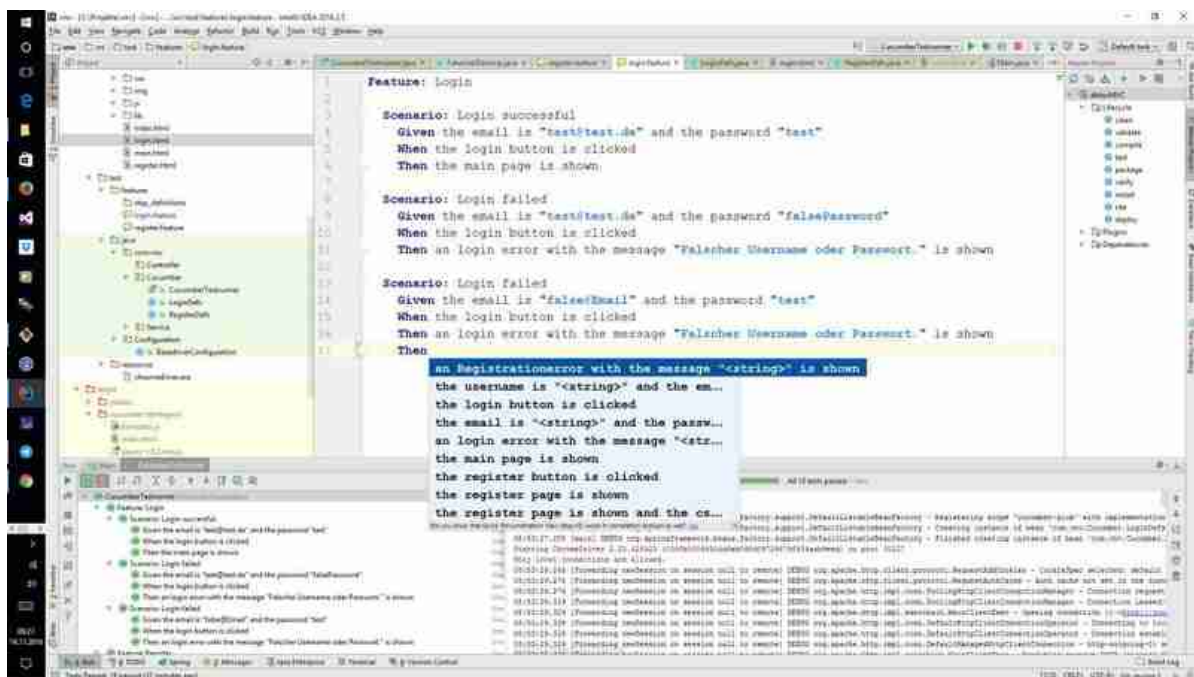
Hi,

today we proudly announce that we have register and login successfully running.

Furthermore we configured Cucumber this week, so we can do easily frontend testing.

We've written Cucumber tests for register and login. You can find the .feature files in the use case descriptions ([1]Register, [2]Login). Generally you can find all Cucumber related files on our [3]GitHub. You can find the results of our Cucumber tests [4]here.

In the following screenshot you can see that our IDE supports syntax highlighting and autocompletion.



We also made a video, where you can see Cucumber running.

<https://youtu.be/DirBzx-qo2o>

We also want to share the relevant parts of our pom.xml so it should be easier for you if you also use maven. You should add the following dependencies:

```
<dependency>
  <groupId>info.cukes</groupId>
  <artifactId>cucumber-spring</artifactId>
  <version>1.2.4</version>
  <scope>test</scope>
</dependency>
```

```
<dependency>
  <groupId>info.cukes</groupId>
  <artifactId>cucumber-junit</artifactId>
  <version>1.2.4</version>
```

```
<scope>test</scope>
</dependency>
```

```
<dependency>
  <groupId>org.seleniumhq.selenium</groupId>
  <artifactId>selenium-java</artifactId>
  <version>3.0.1</version>
</dependency>
```

1. https://github.com/WMerk/vnvDoc/blob/master/doc/UC_Register.MD
2. https://github.com/WMerk/vnvDoc/blob/master/doc/UC_Login.MD
3. <https://github.com/WMerk/vnv/tree/master/src/test>
4. <https://cdn.rawgit.com/WMerk/vnv/master/src/test/features/Reports/cucumber-htmlreport/index.html>

SquadIT (2016-11-14 15:41:30)

Hey VNV-Team, For each UC you have all the cases in your .feature files. Also you have the screenshots and activity diagrams. Your IDE looks great in colouring etc.. Also you link to everything from your UC. SquadIT

Team Project Ortus (2016-11-15 13:42:55)

Hi VNV Team, your testing looks good and seems to work flawlessly. Thanks for sharing all relevant information in your Github repo, our chief developer Koray had a look at it. We could find your use cases in your .feature files. Your IDE Syntax Highlighting and Colouring also looks good! Keep up the great work and regards from Team Project Ortus

scalaqua (2016-11-22 20:22:25)

Hey VNV, Nothing more to say :awesome: The fact, that you made a video about your testingtool in action is a splendid idea. Greetings TextVenturer

Midterm Handin – VNV-Project (2016-12-11 15:46:15)

[...] Cucumber (blog post) [...]

Final – VNV-Project (2017-06-19 09:43:59)

[...] Cucumber (blog post) [...]

Class Diagram (2016-11-21 10:58)

Hi fellows,

today we wanna share our Class diagram with you. The UML diagram shows the current state. There might be added a few classes in the future.

<https://raw.githubusercontent.com/WMerk/vnvDoc/master/doc/UML/UML.png>

We generated our UML diagram with the integrated diagram-tool from IntelliJ.

Additionally we share a few thoughts on our database structure: Using a NoSQL database, we don't have a real database schema. We store Objects of our different Entities. At the current state we have the following Entities: User, Message, Post and Category. You can see such a Entity class below. The fields are annotated with Id, Attribute or Reference, depending on the meaning of the field. Optionally some fields are annotated with Indexed, if we need an index for searching on this fields.

https://raw.githubusercontent.com/WMerk/vnvDoc/master/doc/UML/DB_Post.PNG

Furthmore we implemented two more use cases this week. We are happy to pronounce that you are now able to create a new offer and you could delete your account on vnv (of course you won't).

gtagroup (2016-11-22 10:44:55)

Hey there, your class diagram looks very impressive compared to the one we created. It is also nice that you used multiple arrows for different dependencies. Well done! Lovely greetings <3 GTA-Group

Team Ortus (2016-11-22 10:49:40)

Hi, the UML created with IntelliJ looks impressive and very detailed. Keep it up! Greetings, Project Ortus Team

Midterm Handin - VNV-Project (2016-12-11 15:46:20)

[...] Class Diagram (blog post) [...]

Final - VNV-Project (2017-06-19 09:44:07)

[...] Class Diagram (blog post) [...]

MVC Architecture (2016-11-26 23:11)

Hey,

this week we wrote our Software Architecture Document, which you can find [1]here. Feel free to check our interesting architecture using Spring and AngularJS. We also redrew our class diagram. You can find the new ordered version below or in the SAD (or GitHub).

https://raw.githubusercontent.com/WMerk/VnVProject/master/doc/UML/classdiagram_dep_marked.png

There's also the MVVM Architecture, the one we're using.

<https://raw.githubusercontent.com/WMerk/vnvDoc/master/doc/SAD/mvvm.png>

We also worked on updating user information, which is now possible and social login (might come in nearby future).

1. <https://github.com/WMerk/vnvDoc/blob/master/doc/SAD.MD>

Team Project Ortus (2016-11-28 20:19:41)

Good evening Team VNV, your SAD looks complete and is pretty easy to understand. We think you have everything in there we currently need to know. Also we like the new version of our class diagram, nice colours! Maybe we can get a link to see a live version of VNV? Keep on guys! Regards, Team Project Ortus

vnvproject (2016-11-28 20:31:41)

Hi TeamOrtus, we love to hear that you like the new version of your class diagram ;) Of course you can find vnv running here: <https://dhw-projekt.data.kit.edu> Cheers vnv

Team Mathinator (2016-11-28 20:44:52)

Hi, as always, you seem to have outdone yourselves! The new version of the UML diagram is clean, organized and easy to grasp. The SAD is also self-explanatory and well written. Nicely done! So long, Team Mathinator

squaditblog (2016-12-07 10:47:39)

Hi there :) Your SAD seems to be complete and your MVVM architecture diagram looks really good. Keep up the good work, SquadIT

Midterm Handin – VNV-Project (2016-12-11 15:46:23)

[...] MVC Architecture (blog post) [...]

Final – VNV-Project (2017-06-19 09:44:09)

[...] MVC Architecture (blog post) [...]

1.3 December

Gantt Chart (2016-12-04 15:52)

Hi,

This week we worked on our Gantt Chart showing all the work we done so far. Using YouTrack we can generate a Gantt Chart directly in YouTrack. You can find that Gantt Chart [1]here. However you can't see the tags in this Chart. So we had to export all the issues to MS Project. Therefore we wrote a parser to translate from "YouTrack csv" to "MS Project csv". You can find the result below. First there is a detailed version and second a version where don't split the tags into single issues. You can find the parser [2]here, if you interested.

<https://raw.githubusercontent.com/WMerk/vnvDoc/master/doc/gantt/result.png>

Second (a bit) less detailed version:

https://raw.githubusercontent.com/WMerk/vnvDoc/master/doc/gantt/result_tag.png

From our point of view all of these Charts are not very good at telling us how long we spent on what task or type. YouTrack can generate time reports, where you can see how long you spent on which issue and what you were doing. We think that this is quite a good solution for time management. You can find the result on our [3]GitHub repo.

Cheers vnv

1. <https://vnv.myjetbrains.com/youtrack/reports/gantt/109-4?view=actual>

2. <https://github.com/WMerk/YTG>

3. <https://github.com/WMerk/vnvDoc/tree/master/doc/time>

Team Project Ortus (2016-12-05 17:29:26)

Hi VNV, you did a great job like always. Your Gant Chart / Diagramm is easy to understand and looks clearly sturctured. Thanks again for helping us with parsing the Data generated from YouTrack!! Keep going and regads, Team Project Ortus

The Mathinator (2016-12-06 22:47:33)

Hey VNV, Your Parser, which converte the YouTrack.csv to MS Project.csv works fine. All Issues were implemented and the Gant Chart is done correctly. Well done guys. By the way we used your Parser too, so thank you one more time, for a great job. Mathinator

Midterm Handin - VNV-Project (2016-12-11 15:46:26)

[...] Gantt Chart (blog post) [...]

Final - VNV-Project (2017-06-19 09:44:11)

[...] Gantt Chart (blog post) [...]

Midterm Handin (2016-12-11 15:45)

Hi everyone,

today we wanna sum up all the work we did so far. You can find all the important links below:

GitHub repos:

- [1]GitHub (Documentation)
- [2]GitHub (Code)

YouTrack:

- [3]Issues
- [4]Agile Board
- [5]Time Report

Project Vision:

- [6]Project Vision (blog post)

Project Roles:

- [7]Project Roles (blog post)

SRS:

- [8]SRS
- [9]OUCD
- [10]Software Requirements Specification (blog post)

Use Cases:

- Use Case Discriptions:
 - [11]UCD Register
 - [12]UCD Login
 - [13]UCD Change password
 - [14]UCD Edit profile

- [15]UCD Create new offer
- [16]UCD Add friend
- [17]UCD Delete Account
- [18]Use Cases (blog post)

Scrum:

- [19]Scrum (blog post)
- See YouTrack (above)
- Burndowns:
 - [20]Elab #1
 - [21]Elab #2
 - [22]Constr #1
 - [23]Constr #2
 - [24]Constr #3

Testing:

- [25]Cucumber (blog post)
- [26]Cucumber (video)
- [27]Cucumber (feature files)
- [28]Cucumber test results
- [29]Unittests result (coverage)

Architecture:

- [30]Class diagram (full)
- [31]Class Diagram (reduced)
- [32]Class Diagram (blog post)
- [33]SAD
- [34]MVVM Architecture
- [35]MVC Architecture (blog post)

Long-term planning:

- [36]Gantt Chart (blog post)
- [37]Time Report

Demo:

- [38]vnn runs here

Presentation:

- [39]presentation slides

1. <https://github.com/WMerk/vnnDoc>
2. <https://github.com/WMerk/vnn>
3. <https://vnn.myjetbrains.com/youtrack/issues>
4. <https://vnn.myjetbrains.com/youtrack/agiles/>
5. <https://vnn.myjetbrains.com/youtrack/reports/time/125-10>
6. <https://vnnproject.wordpress.com/2016/10/11/project-vision-of-vnn/>
7. <https://vnnproject.wordpress.com/2016/10/12/project-roles/>
8. <https://github.com/WMerk/vnnDoc/blob/master/doc/SRS.MD>
9. <https://raw.githubusercontent.com/WMerk/vnnDoc/master/doc/use%20cases/SRS.png>
10. <https://vnnproject.wordpress.com/2016/10/24/software-requirements-specification/>
11. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_Register.MD
12. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_Login.MD
13. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_ChangePassword.MD
14. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_EditProfile.MD
15. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_CreateNewOffer.MD
16. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_AddFriend.MD
17. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_DeleteAccount.MD
18. <https://vnnproject.wordpress.com/2016/10/31/use-cases/>
19. <https://vnnproject.wordpress.com/2016/11/03/scrum/>
20. <https://vnn.myjetbrains.com/youtrack/agiles/84-2/85-8?chart>
21. <https://vnn.myjetbrains.com/youtrack/agiles/84-2/85-9?chart>
22. <https://vnn.myjetbrains.com/youtrack/agiles/84-2/85-10?chart>
23. <https://vnn.myjetbrains.com/youtrack/agiles/84-2/85-11?chart>
24. <https://vnn.myjetbrains.com/youtrack/agiles/84-2/85-21?chart>
25. <https://vnnproject.wordpress.com/2016/11/14/cucumber/>
26. <https://youtu.be/DirBzx-qo2o>
27. <https://github.com/WMerk/vnn/tree/master/src/test/features>
28. <https://cdn.rawgit.com/WMerk/vnn/master/src/test/features/Reports/cucumber-htmlreport/index.html>
29. <https://cdn.rawgit.com/WMerk/vnnDoc/master/testReport/unittests/index.html>
30. https://raw.githubusercontent.com/WMerk/vnnDoc/master/doc/UML/classdiagram_dep_marked.png
31. <https://raw.githubusercontent.com/WMerk/vnnDoc/master/doc/UML/small.png>

32. <https://vnpvproject.wordpress.com/2016/11/21/class-diagram/>
 33. <https://github.com/WMerk/vnvDoc/blob/master/doc/SAD.MD>
 34. <https://raw.githubusercontent.com/WMerk/vnvDoc/master/doc/SAD/mvvm.png>
 35. <https://vnpvproject.wordpress.com/2016/11/26/mvc-architecture/>
 36. <https://vnpvproject.wordpress.com/2016/12/04/gantt-chart/>
 37. <https://vnpv.myjetbrains.com/youtrack/reports/time/125-10>
 38. <https://dhw-projekt.data.kit.edu/>
 39. https://drive.google.com/open?id=0B_hUkn0i7I7GamgybGJhOW1fRGM
-

2. 2017

2.1 April

We are getting started again (2017-04-17 19:09)

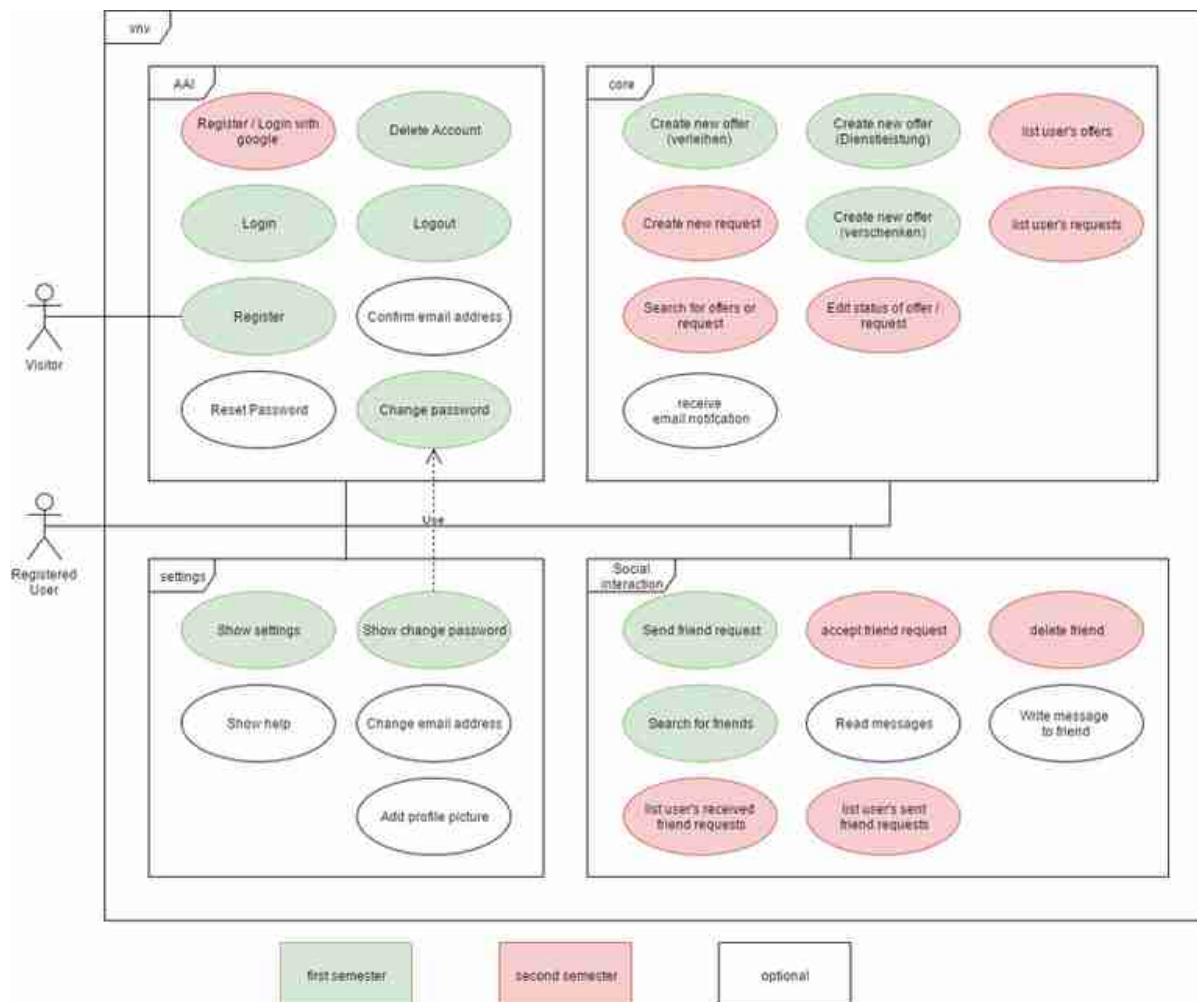
The new semester started and so we are getting started again. We defined the project scope for the following months. In the OUCD below you can see the use cases we will implement in the next time colored in red. We wrote the Use Case Descriptions for these use cases and you can find them on our [1]github repo. We are very proud to announce that we already implemented one of these use cases. Therefore it is now possible to register and login with a google account.

We also checked how much time we spent on the use cases we implemented in the last semester. You can find all of the time reports at our [2]youtrack and the relevant links also in [3]this document.

Furthermore we thought about risks and made a risk plan, that can be found [4]here

If you have any questions left, feel free to put them into the comment section.

Cheers vnv



1. <https://github.com/WMerk/vnvDoc/tree/master/doc>
2. <https://vnv.myjetbrains.com/youtrack/reports>
3. <https://github.com/WMerk/vnvDoc/blob/master/doc/Time%20Reports.MD>
4. <https://github.com/WMerk/vnvDoc/blob/master/doc/risks/Risk%20Management.pdf>

bollo157 (2017-04-25 09:04:56)

Hi VNV, nice to hear again from you. We are looking forward to your new use cases, especially the "delete friend" function :P Your first link to youtrack (<https://vnv.myjetbrains.com/youtrack/reports>) does not work. I am getting an error message which says I am not allowed to create reports. The other time reports look good. Your risk management also looks good and well written. It is easy to understand where you guys see the risk of your projects. Keep going the good work! Regards, Andreas - Project Ortus

vnvproject (2017-04-25 10:33:51)

Hi Andreas, thanks for your comment on the link. We already noticed this, but it actually works. On the right it says, that you are not allowed to create new reports, but on the left there is a button "more" where you can find more reports. And you are allowed to look at them.

Sascha Hug (2017-04-25 09:14:06)

Hi VNV-Team, your information about the last semester looked very reliable. The Dokumentation about your vision and the following Use Cases looks very demanding. Your overview is clearly structured. We wish you a good start in the new semester and keep going. Best regards Mathinator

energycheetah (2017-04-25 09:15:15)

Hey, nice work here. I think there is nothing to complain. Looks that you have big plans for this semester, do you think you can finish this on time? Cheers, Cheetah

vnvproject (2017-04-25 10:36:11)

Hi Cheetah, thanks for your comment. We are very confident to be able to finish in time. We already almost finished all the "social interaction" use cases from the scope and we will finish the other use cases in time. Cheers, vnv

Final – VNV-Project (2017-06-19 09:44:15)

[...] Risk Management (blogpost) [...]

2.2 May

Function points (2017-05-01 18:38)

Last week we had a look at function points. We calculated function points for all our use cases, including the new ones. Evaluating function points and the spent time on the already finished use cases, we can predict the time we will need for the other use cases.

For the calculation we used a excel sheet. The complexity of EOs, EQs, ... are defined by the number of FTRs, RETs and DETs. You can find the result of these calculations [1]here.

You might notice that the time spent for Documentation and also Testing does not really scale with the amount of Function point, but just the time spent coding.

We also did the Complexity Adjustment Table for our project and you can see the result below.

Complexity Adjustment Table

ITEM	COMPLEXITY ADJUSTMENT QUESTIONS	SCALE						
		No Influence						Essential
		0	1	2	3	4	5	
1	Does the system require reliable backup and recovery?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
2	Are data communications required?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
3	Are there distributed processing functions?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
4	Is performance critical?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5	Will the system run in an existing, heavily utilized operational environment?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
6	Does the system require on-line data entry?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
7	Does the on-line data entry require the input transaction to be built over multiple screens or operations?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
8	Are the master files updated on-line?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
9	Are the inputs, outputs, files or inquiries complex?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
10	Is the internal processing complex?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
11	Is the code to be designed reusable?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
12	Are conversion and installation included in the design?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
13	Is the system designed for multiple installations in different organizations?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
14	Is the application designed to facilitate change and ease of use by the user?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	

[Domain Characteristic Table](#) | [FP Calculation](#)

Cheers,

vnv

1. <https://github.com/WMerk/vnvDoc/blob/master/doc/FP.pdf>

bollo157 (2017-05-02 08:52:38)

hi vnv, your fp calculation is very detailed and well done. you've included almost every detail we can imagine. it would be interesting to know if your predicted time for new / feature uses cases will match with the real time you will spend... keep us updated! regards Andreas - Project Ortus

SquadIT (2017-05-02 08:55:29)

Hey VNV, your Complexity Adjustment Table looks good, as well as your Excel sheet with your calculations. It is very detailed. The time over FP diagram is also very detailed but maybe you could have split the information up into two diagrams? There is very much information in it, which makes it hard to find data for only development (e.g.) Keep going the good work, greetings SquadIT

Final - VNV-Project (2017-06-19 09:44:13)

[...] Function points (blogpost) [...]

Test plan (2017-05-07 18:10)

Today we want to write about testing: Ever since we started with the project, we wrote unittests for the implemented code. You can find the result of these tests [1]here. If you click through it, you also can find the coverage marked in the implemented code. The test code of our unittests can be found [2]here.

Our pom.xml file with the relevant stuff for testing (including fakers) is on our [3]github.

Our test plan can be found [4]here.

Cheers,

vnv

1. <https://cdn.rawgit.com/WMerk/vnvDoc/master/testReport/unittests/index.html>
2. <https://github.com/WMerk/vnv/tree/master/src/test/java/com/vnv>
3. <https://github.com/WMerk/vnv/blob/master/pom.xml>
4. <https://github.com/WMerk/vnvDoc/blob/master/doc/TestPlan.MD>

bollo157 (2017-05-08 21:23:00)

hi team vnv, thanks for sharing your testing and experience with us! you seem to have done great work as almost every week! according to your documents the tests are running fine, does not matter if they are passing or not. i think your testing document is not finished yet - just like ours :P keep going - we will check back! regards, andreas - project ortus

Sascha Hug (2017-05-09 08:58:59)

Hi vnv, your Test Plan looks great and completed. It seems, you really know what you are doing and what tools could be the best choice for your projekt. We are really excited to hear about your results in several weeks. Also I like your Test Coverage result. Keep us up to date! Regards Mathinator Team

AnKaLu Team (2017-05-30 09:25:44)

Hi Team VNV, everything looks great und completed. The only point we would like to see is the updated testing document with your names corresponding to your Roles in "People and Roles". Best regards, AnKaLu-Team

Final - VNV-Project (2017-06-19 09:43:58)

[...] Test plan (blogpost) [...]

Refactoring (2017-05-15 10:12)

Hi,

this week we refactored a example repository. Therefore, every team member created his own repository:

- [1]Waldemar
- [2]Patrick
- [3]Gabriel

Cheers vnv

1. <https://github.com/WMerk/WMFowler>
2. <https://github.com/ptreyer/Fowler>
3. <https://github.com/zachmann/Fowler>

energycheetah (2017-05-16 08:43:18)

Nice homework, guys. Maybe Waldemar needs to make some more commits, so that he has minimum 7 ;) Cheers Cheetah

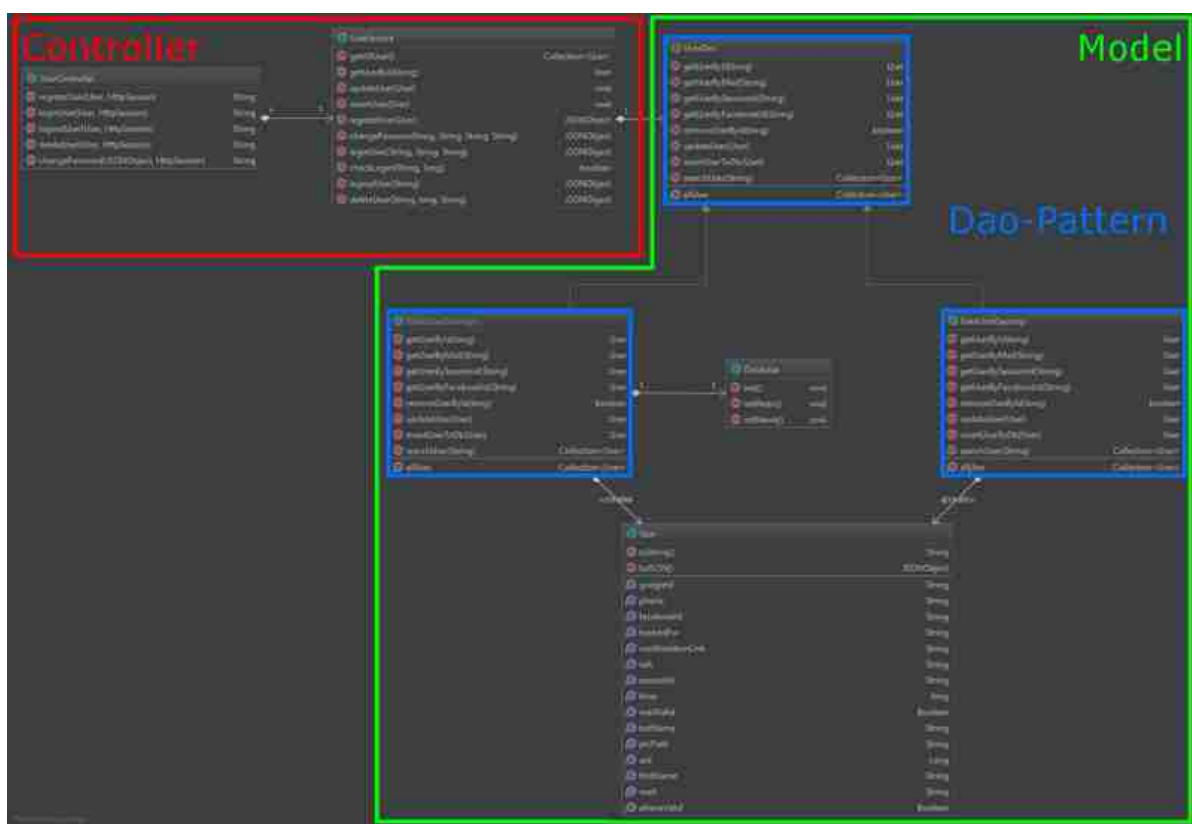
Team Ortus (2017-05-16 09:21:56)

Hi, well made. I bet your IDE saved you some unnecessary work time. Cheers, Team Ortus

Hi,

this week we want to share our thoughts on pattern. Ever since we started we used the DAO-pattern (Data Access Object). The DAO provides an abstract interface to the database. In this way we achieve a simple separation between business logic and persistence logic. The advantage of this is, that we can change each of this parts independently of each other, because changes to persistence logic do not affect business logic as long as the interface remains correctly implemented; changes to business logic can rely on the interface, respectively.

Another advantage is the easy use of multiple databases. We use different databases for production and developing. Using the DAO-pattern, we can easily switch between databases because we provide implementation for both and can rely on the interface.



It was our own decision to use this pattern so obviously we would use this pattern also if we didn't have to, but we want to note that there might be potential code duplications due to this pattern and if you use just one database and don't have to switch between them easily, this pattern might feel a bit overpowered. however for our needs it is very fitting.

Cheers vnv.

energycheetah (2017-05-23 08:24:36)

Hey, nice pattern.. Maybe you could add a comparison to the MVC picture without your pattern.. Cheers Cheetah

joinsportsblog (2017-05-23 09:26:02)

Hi vnvproject, i think you have choosen a good solution for your application with this pattern. So you have a clear separation between business logic and persistence logic and in case theres a change in future it will be easy to modify the certain part. Greetings, the JoinSports-Team

TeamOrtus (2017-05-29 23:32:06)

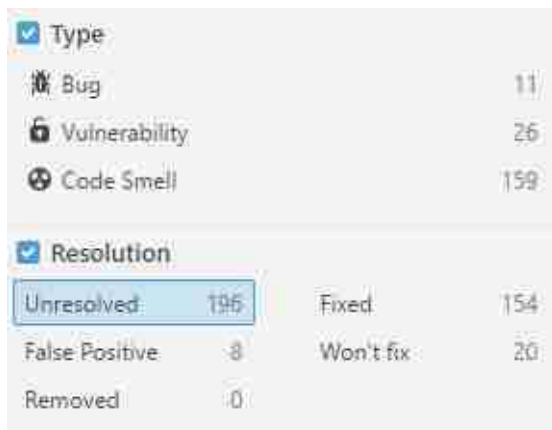
Hi VNV, as other projects also said, we like your pattern. It is clearly structured and easy to understand. Regards Team Ortus

Metrics (2017-05-29 16:54)

Hi there,

metrics can help to ensure code quality.

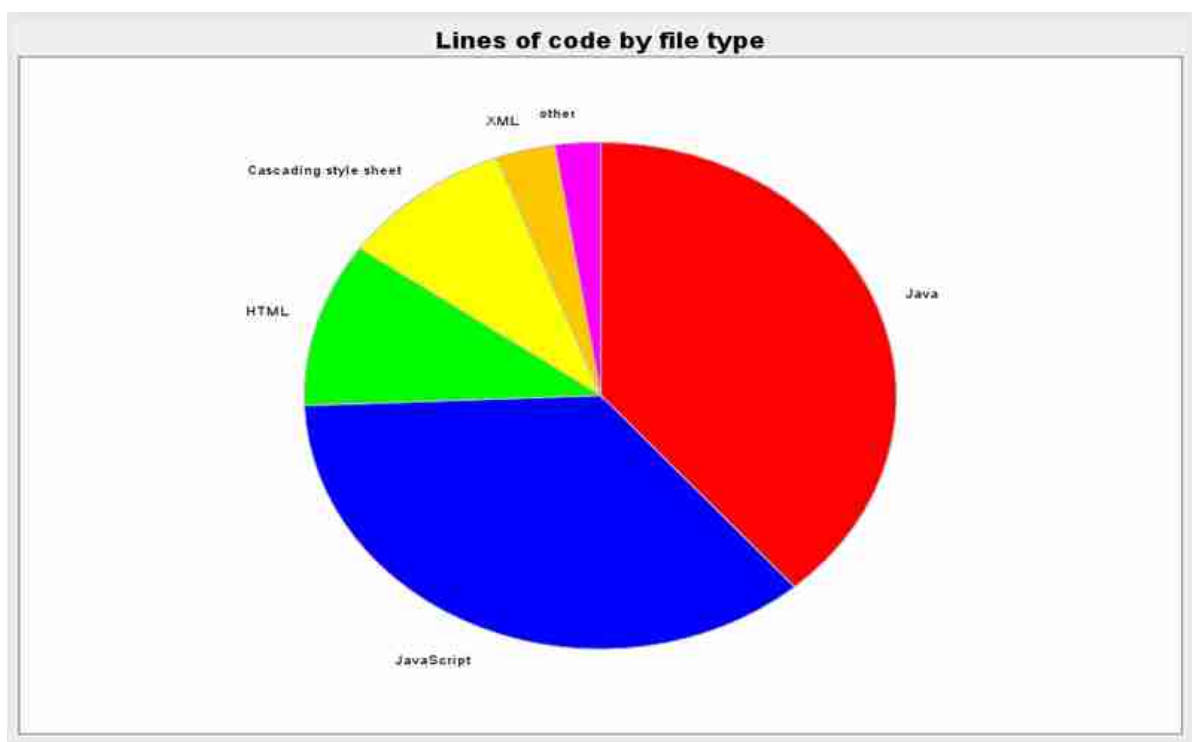
You can find some of our metrics on [1]sonarqube. You can see from the following screenshot, that we did fix some issues with our code, but there were also some false positive issues and some that are not fixworthy for us. An example of a false positive issue is [2]this one.

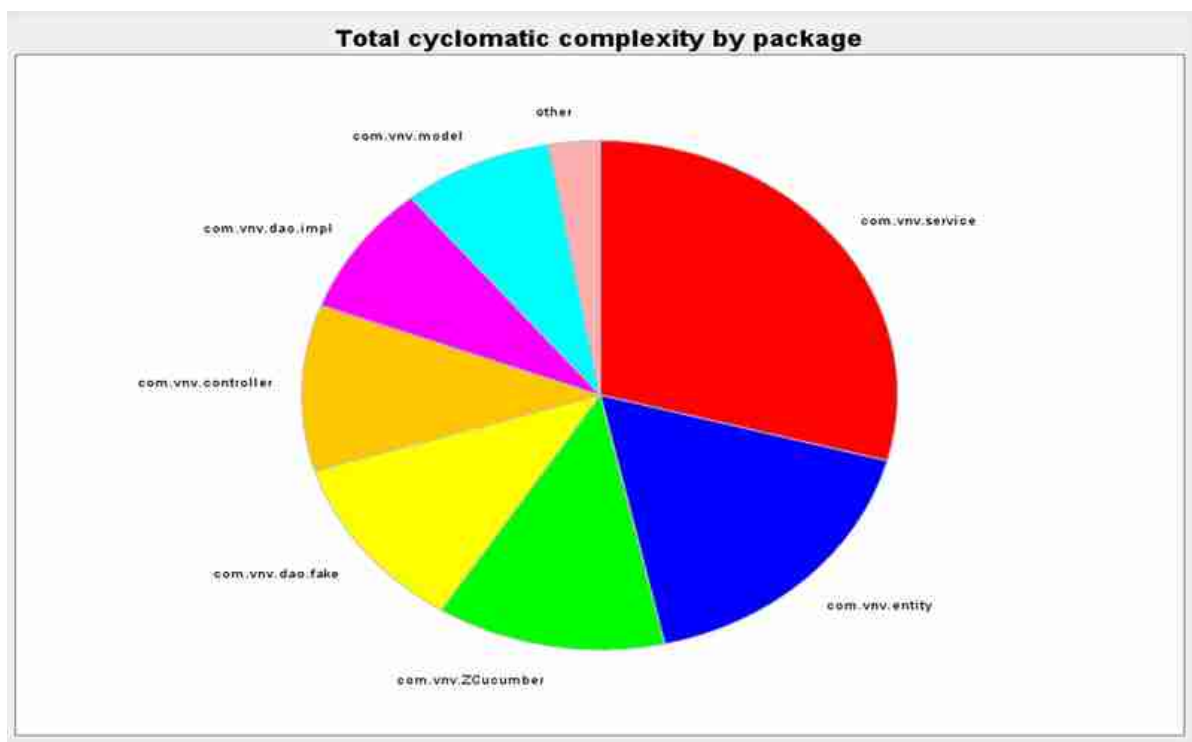
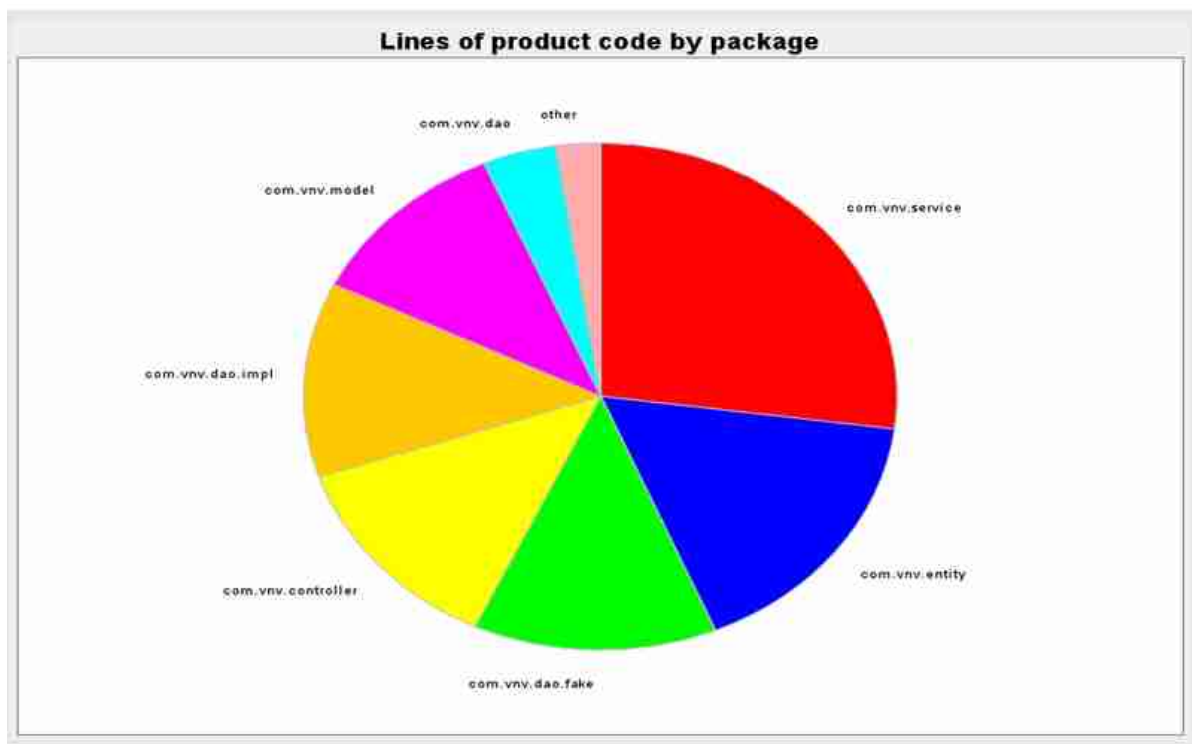


You can find some of our refactoring in [3]this and [4]this commit.

For test coverage we use the build-in tool from IntelliJ. Unfortunately it is not possible to integrate this tool with sonarqube. That's the reason why our coverage result on sonarqube is so bad. However, you can find our Unittest coverage report [5]here.

Additionally, you can find some metrics by IntelliJ [6]here or in the added pictures below.





Cheers vnv.

1. <https://sonarqube.com/dashboard?id=de.dhbw.ka.vnv%3Avnvproject>
2. <https://sonarqube.com/issues#issues=AVvsMlkbtcYCvBl46LR>
3. <https://github.com/WMerk/vnv/commit/94e73012fed8f9000e84ad1e71bde7f05ffb28b3>

4. <https://github.com/WMerk/vnv/commit/35d75e321a8e9fa4d533d1173a3e8c8b1731758b>
5. <https://cdn.rawgit.com/WMerk/vnvDoc/master/testReport/unittests/index.html>
6. <https://github.com/WMerk/vnvDoc/tree/master/doc/metrics>

Tim (2017-05-29 23:32:09)

Hey Team VNV, very nice refactored code in the second commit 51 files changed 281 lines of refactored code! 40-50 % overall unittest coverage on a project like this is good beginning.

TeamOrtus (2017-05-29 23:41:00)

Hi VNV, your work is very clean. We would like to have a little more detail about your false positives (just a short description: why is it a false positive? Why are you cool with it? ...). All in all nice work. Keep it up. Regards, Team Ortus
<http://projectortus.abuchmann.de/>

vnvproject (2017-05-30 08:48:15)

Hi Team Ortus, for example it complained that there was a hardcoded password, but it wasn't a password. You also can checkout these issues on our sonarqube by yourself, if you want to know any more. Regards vnv

Final - VNV-Project (2017-06-19 09:44:06)
[...] Metrics (blogpost) [...]

2.3 June

Test Coverage (2017-06-12 12:39)

Hi,

we might have mentioned it before, but we are using unittests and have a test coverage. you can find a coverage report [1]here. For this report we used the tool integrated in IntelliJ. Additionally we now use JaCoCo for coverage reports and integrated them to our [2]Sonarqube Board. So you can find the current coverage report [3]there.

You can find more information in our [4]Testplan.

Cheers vnv

1. <https://cdn.rawgit.com/WMerk/vnvDoc/master/testReport/unittests/index.html>
2. <https://sonarcloud.io/dashboard?id=de.dhbw.ka.vnv%3Avnvproject>
3. https://sonarcloud.io/component_measures/metric/coverage/tree?id=de.dhbw.ka.vnv%3Avnvproject
4. <https://github.com/WMerk/vnvDoc/blob/master/doc/TestPlan.MD>

Final - VNV-Project (2017-06-19 09:44:04)
[...] Test Coverage (blogpost) [...]

Continuous Integration (2017-06-12 14:41)

Hi,

We use TravisCI for Continuous Integration. We have integrated the unit tests, code coverage and metrics into this deployment cycle. A push to github triggers tests, building, analysis and deployment to our [1]server.

We also created a [2]Installation Guide, so if you want to deploy it on your own server, you can follow this guide. Thanks to Tobias from Mathinator who tested the Installation process and noticed some minor points in the Installation guide, so we could improve it.

Cheers vnv.

1. <https://dhbw-projekt.data.kit.edu/>

2. <https://github.com/WMerk/vnvDoc/blob/master/doc/Installation.MD>

Tobias / Team Mathinator (2017-06-13 10:03:41)

Hi Team VNV, I followed the installation routine and even though it consists of quite a few steps it worked as described. There were no errors or parts that you didn't cover. Proof of installation: <http://img4host.net/upload//13100207593f-9bff8658d.png> Nice work guys!

Final - VNV-Project (2017-06-19 09:44:18)

[...] Continuous Integration and Installation (blogpost) [...]

Final (2017-06-19 09:43)

Hi everyone,

today we wanna sum up all the work we did so far. You can find all the important links below:

GitHub repos:

- [1]GitHub (Documentation)
- [2]GitHub (Code)

YouTrack:

- [3]Issues
- [4]Agile Board
- [5]Time Report

Project Vision:

- [6]Project Vision (blog post)

Project Roles:

- [7]Project Roles (blog post)

SRS:

- [8]SRS
- [9]OUCD
- [10]Software Requirements Specification (blog post)

Use Cases:

- Use Case Discriptions:
 - [11]UCD Register
 - [12]UCD Login
 - [13]UCD Register / Login with Google
 - [14]UCD Change password

- [15]UCD Edit profile
- [16]UCD Create new Offer
- [17]UCD Create new Request
- [18]UCD List Offers
- [19]UCD List Requests
- [20]UCD Search through offers and requests
- [21]UCD Edit Status
- [22]UCD Add friend
- [23]UCD List Received Friendrequests
- [24]UCD List Sent Friendrequests
- [25]UCD Accept Friend Request
- [26]UCD Delete Friend
- [27]UCD Delete Account
- [28]Use Cases (blog post)

Scrum:

- [29]Scrum (blog post)
- See YouTrack (above)
- Burndowns: (Unfortunately Youtrack does not mark the spent time when it actually is spent, but when the issue is closed, the whole amount of spent time is marked as a whole)
 - [30]Elab #1 (nice)
 - [31]Elab #2 (nice)
 - [32]Constr #1
 - [33]Constr #2
 - [34]Constr #3
 - [35]Constr #4
 - [36]Elab #3
 - [37]Constr #5
 - [38]Constr #6
 - [39]Constr #7
 - [40]Constr #8
 - [41]Constr #9
 - [42]Constr #10

Testing:

- [43]Testplan
- [44]Test plan (blogpost)

- [45]Cucumber (blog post)
- [46]Cucumber (video)
- [47]Cucumber (feature files)
- [48]Cucumber test results
- [49]Unittests result (coverage)
- [50]Unittest coverage in sonarqube
- [51]Test Coverage (blogpost)
- [52]Sonarqube
- [53]Metrics (blogpost)

Architecture:

- [54]Class diagram (full)
- [55]Class Diagram (reduced) with Dao-Pattern marked
- [56]Class Diagram (blog post)
- [57]SAD
- [58]MVVM Architecture
- [59]MVC Architecture (blog post)

Long-term planning:

- [60]Gantt Chart (blog post)
- [61]Time Report
- [62]Time Reports for each Use case
- [63]Function points calculation
- [64]Function points (blogpost)
- [65]Risk Management
- [66]Risk Management (blogpost)

Installation:

- [67]Installation guide
- [68]Continuous Integration and Installation (blogpost)

Demo:

- [69]vnn runs here

Presentation:

- [70]midterm presentation slides
- [71]final presentation slides

1. <https://github.com/WMerk/vnnDoc>
2. <https://github.com/WMerk/vnn>
3. <https://vnn.myjetbrains.com/youtrack/issues>
4. <https://vnn.myjetbrains.com/youtrack/agiles/>
5. <https://vnn.myjetbrains.com/youtrack/reports/time/125-11>
6. <https://vnnproject.wordpress.com/2016/10/11/project-vision-of-vnn/>
7. <https://vnnproject.wordpress.com/2016/10/12/project-roles/>
8. <https://github.com/WMerk/vnnDoc/blob/master/doc/SRS.MD>
9. <https://raw.githubusercontent.com/WMerk/vnnDoc/master/doc/use%20cases/SRS2.png>
10. <https://vnnproject.wordpress.com/2016/10/24/software-requirements-specification/>
11. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_Register.MD
12. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_Login.MD
13. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_RegisterLoginGoogle.MD
14. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_ChangePassword.MD
15. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_EditProfile.MD
16. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_CreateNewOffer.MD
17. https://github.com/WMerk/vnnDoc/tree/master/doc/UC_CreateNewRequest.MD
18. https://github.com/WMerk/vnnDoc/tree/master/doc/UC_ListOffers.MD
19. https://github.com/WMerk/vnnDoc/tree/master/doc/UC_ListRequests.MD
20. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_SearchOffersRequests.MD
21. https://github.com/WMerk/vnnDoc/tree/master/doc/UC_EditStatus.MD
22. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_AddFriend.MD
23. https://github.com/WMerk/vnnDoc/tree/master/doc/UC_ListReceivedFriendRequests.MD
24. https://github.com/WMerk/vnnDoc/tree/master/doc/UC_ListSentFriendRequests.MD
25. https://github.com/WMerk/vnnDoc/tree/master/doc/UC_AcceptFriendRequest.MD
26. https://github.com/WMerk/vnnDoc/tree/master/doc/UC_DeleteFriend.MD
27. https://github.com/WMerk/vnnDoc/blob/master/doc/UC_DeleteAccount.MD
28. <https://vnnproject.wordpress.com/2016/10/31/use-cases/>
29. <https://vnnproject.wordpress.com/2016/11/03/scrum/>
30. <https://vnn.myjetbrains.com/youtrack/agiles/84-2/85-8?chart>

31. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-9?chart>
32. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-10?chart>
33. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-11?chart>
34. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-21?chart>
35. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-25?chart>
36. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-26?chart>
37. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-27?chart>
38. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-28?chart>
39. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-29?chart>
40. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-30?chart>
41. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-31?chart>
42. <https://vnnv.myjetbrains.com/youtrack/agiles/84-2/85-32?chart>
43. <https://github.com/WMerk/vnnvDoc/blob/master/doc/TestPlan.MD>
44. <https://vnnvproject.wordpress.com/2017/05/07/test-plan/>
45. <https://vnnvproject.wordpress.com/2016/11/14/cucumber/>
46. <https://youtu.be/DirBzx-qo2o>
47. <https://github.com/WMerk/vnnv/tree/master/src/test/features>
48. <https://cdn.rawgit.com/WMerk/vnnv/master/src/test/features/Reports/cucumber-htmlreport/index.html>
49. <https://cdn.rawgit.com/WMerk/vnnvDoc/master/testReport/unittests/index.html>
50. https://sonarcloud.io/component_measures/metric/coverage/tree?id=de.dhbw.ka.vnnv%3Avnnvproject
51. <https://vnnvproject.wordpress.com/2017/06/12/test-coverage/>
52. <https://sonarcloud.io/dashboard?id=de.dhbw.ka.vnnv%3Avnnvproject>
53. <https://vnnvproject.wordpress.com/2017/05/29/metrics/>
54. https://raw.githubusercontent.com/WMerk/vnnvDoc/master/doc/UML/current_marked.png
55. <https://raw.githubusercontent.com/WMerk/vnnvDoc/master/doc/UML/small2Pattern.png>
56. <https://vnnvproject.wordpress.com/2016/11/21/class-diagram/>
57. <https://github.com/WMerk/vnnvDoc/blob/master/doc/SAD.MD>
58. <https://raw.githubusercontent.com/WMerk/vnnvDoc/master/doc/SAD/mvvm.png>
59. <https://vnnvproject.wordpress.com/2016/11/26/mvc-architecture/>
60. <https://vnnvproject.wordpress.com/2016/12/04/gantt-chart/>
61. <https://vnnv.myjetbrains.com/youtrack/reports/time/125-11>
62. <https://github.com/WMerk/vnnvDoc/blob/master/doc/Time%20Reports.MD>
63. <https://github.com/WMerk/vnnvDoc/blob/master/doc/FP.pdf>
64. <https://vnnvproject.wordpress.com/2017/05/01/function-points/>
65. <https://github.com/WMerk/vnnvDoc/blob/master/doc/risks/Risk%20Management.pdf>
66. <https://vnnvproject.wordpress.com/2017/04/17/we-are-getting-started-again/>
67. <https://github.com/WMerk/vnnvDoc/blob/master/doc/Installation.MD>
68. <https://vnnvproject.wordpress.com/2017/06/12/continuous-integration/>
69. <https://dhbw-projekt.data.kit.edu/>
70. https://drive.google.com/open?id=0B_hUkn0i7I7GamgybGJhOW1fRGM
71. <https://drive.google.com/file/d/1J6VCOMSOGflmqEEN2bQnDlHxMM5CLE8k90D8UcBVie4/view>



BlogBook v0.9,
 \LaTeX 2_ε & GNU/Linux.
<https://www.blogbooker.com>

Edited: June 19, 2017

