



Low quality pictures

nappydevelopment.wordpress.com

Contents

1	2015	5
1.1	September	6
	About (2015-09-28 08:46)	7
	Hello Wordpress (2015-09-28 08:54)	8
	A little bit more about our project vision... (2015-09-30 14:01)	9
1.2	October	11
	Responsibilities and technolgy choice (2015-10-05 13:40)	12
	Software Requirements Specification (2015-10-08 08:10)	15
	UseCase and Mockups (2015-10-23 11:46)	17
1.3	November	19
	Class diagramms and architecture (2015-11-05 14:22)	20
	Software Architecture Document (2015-11-12 09:00)	22
	We are on Jira (2015-11-21 17:13)	23
	Gantt Chart (2015-11-26 12:32)	24
1.4	December	25
	Midterm (2015-12-18 15:23)	26
2	2016	29
2.1	April	30
	Riskmanagement and planned UseCase's (2016-04-10 18:00)	31
	Function points (2016-04-17 17:00)	33
	TDD, JUnit and UC's (2016-04-23 13:37)	34
2.2	May	35
	Fowler's Refactoring (2016-05-01 11:45)	36
	Design Pattern (2016-05-09 09:13)	37
	Metrics (2016-05-29 17:00)	40
	Automatic deploy and Test Plan (2016-05-30 11:30)	42

1. 2015

1.1 September

About (2015-09-28 08:46)

This is an example of a page. Unlike posts, which are displayed on your blog's front page in the order they're published, pages are better suited for more timeless content that you want to be easily accessible, like your About or Contact information. Click the Edit link to make changes to this page or [1]add another page.

1. <https://wordpress.com/page/100321041/new/>

Hello Wordpress (2015-09-28 08:54)

Welcome to our blog! :)

We are Manuel Bothner, Marc Mahler, Marvin Zerulla and Mehmet Ali Incekara.

This is our GitHub-Link: <https://github.com/nappydevelopment>

A little bit more about our project vision... (2015-09-30 14:01)

As already mentioned

we plan to develop a quiz-game named: "Nappy, the ingenious" (preliminary)

We will create it in Java as desktop-application.

To the game principle:

First you choose a character of the tv-series "The Simpsons" and keep it in mind.

Then the virtual player "Nappy" controlled by the program tries to guess your character.

Nappy will ask you questions like "Is your character male?/Is your character a child?/..." to get more and more information about your character.

At one point Nappy will know which character you have chosen and the number of questions will be noted down!

Then you and Nappy switch roles, Nappy chooses a character and you must try to guess it.

For that you can ask questions out of a pre-build list of questions (same questions that Nappy can ask!) and Nappy answers.

When you are sure that you know the character you enter the name or pick it out of a list with all Simpsons characters or something like that.

If your answer is wrong you lose, is it right the program checks who needed fewer questions and that one wins!

That's

the fundamental

gameplay!

We hope you like our idea!

tom433 (2015-10-02 16:31:52)

The idea is great. I have only one question: Why is the KI called "Nappy"? :D An idea is that you can make it a multiplayer game, so two people can match each other. Have a nice day Tom from tohemidevelopment

mehmetaliincekara (2015-10-02 16:45:13)

Hey Tom, why did we call our KI "Nappy"? I really don't know when and why I used the name the first time but it is more than 5 years ago. Actually we have planned this two gamemodes. Maybe if there is time we will think about your idea and a new gamemode. Thank you for your comment Kind regards Ali from nappydevelopment

caro340 (2015-10-04 19:15:23)

Hey nappydevelopment-team, I like your idea, sounds really fun. Are you planning to extend the game with more series? Would love that for series like The Big Bang Theory or Two and a half men :) Best wishes, caro from sapaca

mehmetaliincekara (2015-10-04 19:55:24)

Hello caro, at the moment we are just planning to use 100 characters from The Simpsons. To expand the database with new characters from The Simpsons or other series should not be a big problem. Thank you for your comment. Kind regards Ali from nappydevelopment

1.2 October

Responsibilities and technology choice (2015-10-05 13:40)

Hey :)

we think it is time to tell you our team members with their responsibilities.

We will use the RUP terminology for the team roles.

Discipline
Team member

Business Modeling
Mehmet Ali Incekara

Requirements
Marvin Zerulla

Analysis and Design
Marvin Zerulla

Implementation
nappydevelopment*

Test
Manuel Bothner and Mehmet Ali Incekara

Deployment
Manuel Bothner

Project Management
Mehmet Ali Incekara

Environment
Marc Mahler

Configuration and Change Managment
Marc Mahler

every member of nappydevelopment will implement. More information in the next table.

Implementation
Team member

Frontend
Manuel Bothner

Backend and Database
Marvin Zerulla, Mehmet Ali Incekara and Marc Mahler

For our desktop-application we will use the following technologies:

- Frontend: JavaFx, HTML, CSS
- Backend: Java
- Database: embedded H2 database
- Testing: JUnit and Sikuli
- Build: Gradle

Kind regards
Ali from nappydevelopment

streboparsegyan (2015-10-08 16:23:47)

Hey nappydevelopment-Team you have clearly defined the responsibilities of your single team members using the RUP terminology. Also your choice of technologies shows, that you have intensively discussed this topic to choose the technologies that fit best for your project. Kind regards strebo-Team

Marvin Zerulla (2015-10-12 09:12:06)

Hey Streboparsegyan, thanks for your comment, we tried to make our Choices as clear as possible. Greetings Marvin from nappydevelopment

tom433 (2015-10-11 14:57:35)

Hello team nappydevelopment, you clearly define your team roles and responsibilities of every team member using the RUB terminology. Your chosen technology is also clear defined, so you have the basics to make a great application. Kind regards Tom from tohemidevelopment

mehmetaliincekara (2015-10-12 09:04:59)

Hello Tom, thank you for your comment. It's nice to read that you used the grading criteria. It would help if you ask us why we are using this technologies or tell us alternative technologies. Have a nice day Ali from nappydevelopment

Larissa (2015-10-11 15:06:03)

Hey nappydevelopment- Team, you defined your team roles using RUP terminology very clearly and illustrative. That is really good to get a quick overview. I would like to know if you thought about adding pictures to your game and why you are using the technologies you chose. For example why aren't you using Java Script but JavaFx. Kind regards MathJunkie Team

mehmetaliincekara (2015-10-12 09:12:22)

Hello Larissa, thank you for your comment. You can find pictures of the game (prototype) in the next blog-entry "Planned functions". Thank you for your interest. We will use JavaFX instead of Swing because JavaFX is more modern than Swing. We don't use Javascript because we are building a desktop-application and JavaScript is not applicable. Have a nice day Ali from nappydevelopment

Software Requirements Specification (2015-10-08 08:10)

Hey :)

we want to publish a screenshot of our application "Nappy, the ingenious". The following screenshot shows a (uncompleted) prototype of our client.



[1]

Now to our SRS:

This is our GitHub-Repository for our documents: <https://github.com/nappydevelopment/docs>

This is the link for our SRS: [https://github.com/nappydevelopment/docs/blob/master/pdfs/Software %20Requirements %20Specification.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Software%20Requirements%20Specification.pdf)

This is the link for our overall Use Case Diagram (UCD): [https://github.com/nappydevelopment/docs/blob/master/pdfs/Overall %20Use %20Case %20Diagram.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Overall%20Use%20Case%20Diagram.pdf)

Have a nice Day

Ali and Marvin from nappydevelopment

1. https://nappydevelopment.files.wordpress.com/2015/10/2015-10-08-10_37_22-nappy-the-ingenious.png

jockelmore (2015-10-18 14:43:32)

Hello nappydevelopment, - Use Template -> You have used the template. - Proof of version control (GIT, or by hand on second page) -> You are using GIT for version control. - Blue text has been replaced with actual information regarding project. -> The blue text is completely - Items that are not relevant have been maintained but contain the comment: not applicable (n/a) -> You have used "not applicable (n/a)" for items which are not relevant. - Items that you don't know yet have been marked with: to be determined (tbd) -> You have used "to be determined (tbd)" to mark Items you don't know at the moment. - SRS contains the overall Use Case Diagram (UCD) -> UCD is linked in the SRS. - the UCD is correct (UML with 5 components: actors, system boundary, UC, association, dependencies AND scope marked in grey or color) -> "new game", "game mode 1" and "game mode

2" do not contain verbs ;) Overall it is a well done SRS! Good work! Kind regards, Jan-Eric from saphijaga

mehmetaliincekara (2015-10-18 18:01:17)

Hello Jan-Eric, thank you for sacrificing time to read our SRS. It wasn't noticeable to us that we don't use verbs in our Use Case Diagram. We will improve our overall UCD and update our SRS with the new UCD. Have a nice day, Ali from nappydevelopment

mehmetaliincekara (2015-10-19 07:55:25)

We updated our overall UCD. Thank you :) Have a nice day Ali from nappydevelopment

mypixy (2015-10-19 07:50:01)

Hey Nappies :) we liked that you have used the template. You have replaced the blue text with your own information. Perhaps you have to look at "3.6 Design Constraints" again - we think you have to describe the for example "software languages, software process requirements, prescribed use of developmental tools, architectural and design constraints, purchased components, class libraries, and so on". Furthermore you have marked all not relevant points with "not applicable (n/a)" and points that you don't know yet with "to be determined (tbd)". In your SRS (1.4) is a link to your overall Use Case Diagram (UCD). As Jan-Eric just told you, you have to look at the "verbs" in your UCD again ;) We think it's a good idea to proof your version control with github. Kind Regards Chiara & Sarah from My Pixy

mehmetaliincekara (2015-10-19 08:18:06)

Hello myPixies, first thank you for reading our SRS. You are right and we update our SRS. A new version of our overall UCD is online. Have a nice day, Ali from nappydevelopment

UseCase and Mockups (2015-10-23 11:46)

Hey :)

this blog entry will link all UseCase's and our mockups. They are also linked in our SRS.

Mockups: <https://github.com/nappydevelopment/docs/blob/master/pdfs/Mockups.pdf>

UseCase Gamemode 1: [https://github.com/nappydevelopment/docs/blob/master/pdfs/1 %20UseCase _Gamemode %201.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/1%20UseCase_Gamemode%201.pdf)

UseCase View Wiki: [https://github.com/nappydevelopment/docs/blob/master/pdfs/2 %20UseCase _View _Wiki.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/2%20UseCase_View_Wiki.pdf)

UseCase View Instruction: [https://github.com/nappydevelopment/docs/blob/master/pdfs/3 %20UseCase _View _Instructions.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/3%20UseCase_View_Instructions.pdf)

UseCase Change Settings: [https://github.com/nappydevelopment/docs/blob/master/pdfs/4 %20UseCase _Change _Settings.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/4%20UseCase_Change_Settings.pdf)

UseCase View Info: [https://github.com/nappydevelopment/docs/blob/master/pdfs/5 %20UseCase _View _Info.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/5%20UseCase_View_Info.pdf)

Have a nice day,
Ali from nappydevelopment

samuelp94 (2015-10-24 13:20:57)

Hello Nappydevelopment, your Use Cases look very well, but in your first Use Case (Gamemode 1) you forget to change the default name in the header. Kind regards, Samuel from saphijaga

mehmetaliincekara (2015-10-24 13:54:50)

Hello Samuel, thank you for your comment. Oups, yes you are right. We will update the first UseCase. Have a nice day Ali from nappydevelopment

tom433 (2015-10-26 08:40:38)

Hello nappydevelopment, your use cases looking great. They are structured, clear and easy to understand. You also have use the template and the use cases are linked to your SRS. Great work! :) Kind regards, tom from tohemidevelopment

mehmetaliincekara (2015-10-26 09:21:32)

Hey Tom, thank you for your comment. Have a nice day Ali from nappydevelopment

tom433 (2015-11-02 08:02:55)

Hello nappydevelopment, all of your scenarios agree with your Activity Diagram paths and keywords are highlighting in the screenshots in the documents, good work :) Kind regards, tom from tohemidevelopment

mehmetaliincekara (2015-11-02 09:18:11)

Hello Tom, thank you for reading our document. Maybe next time you can use the grading criteria. Have a nice day, Ali from

nappydevelopment

tom433 (2015-11-04 12:30:18)

Hello Ali, i have used the grading criteria ;) Kind regards, Tom from tohemidevelopment

Spooser | Jani (2015-11-02 09:09:40)

Hello nappydevelopment, Your scenarios are written correctly and agree with your Activity diagram. Well done! Kind regards, Jani from saphijaga

marcmahlerdhbw (2015-11-02 09:19:29)

Hello Jani, Thank you for your feedback on our project. I am glad that you like our work. Kind regards, Marc

1.3 November

Class diagramms and architecture (2015-11-05 14:22)

Hello everyone,

In this blog entry I would like to show you our class diagramm.
In the overall-diagramm I have highlighted the different parts of our architecture.

You can see our overall-diagramm by following this link:
[https://github.com/nappydevelopment/docs/blob/master/svg/Class %20Diagram.svg](https://github.com/nappydevelopment/docs/blob/master/svg/Class%20Diagram.svg)

If you follow the next links, you will get zoomed-in versions of the different parts our architecture:

Control:
[https://github.com/nappydevelopment/docs/blob/master/pdfs/Control _Class _Diagramm.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Control_Class_Diagramm.pdf)

View:
[https://github.com/nappydevelopment/docs/blob/master/pdfs/View _Class _Diagramm.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/View_Class_Diagramm.pdf)

Model:
[https://github.com/nappydevelopment/docs/blob/master/pdfs/Model _Class _Diagramm.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Model_Class_Diagramm.pdf)

and Database:
[https://github.com/nappydevelopment/docs/blob/master/pdfs/Datenbank _Class _Diagramm.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Datenbank_Class_Diagramm.pdf)

Please be aware, that this might be not the final version. We are currently working on our architecture and it might happen, that things change.

We appreciate your comments.

Have a nice day,

Marc from nappydevelopment

tom433 (2015-11-09 09:22:40)

Hello Nappydevelopment, you use correct UML and your class diagram shows the important methods but no attributes, please add those to the diagram. You have definitely more than 5 classes and that classes are needed for all five use cases promised for Christmas. Good work :) One more thing: On the class diagram overview I can't read the name and some methods of two classes because there is something in front of them. Please change this :) Kind regards, Tom from tohemidevelopment

marcmahlerdhbw (2015-11-09 09:35:06)

Hello Tom, Thank you very much for your comment on our latest post. I will change the diagramm according to your suggestions and try reuploading the diagramm as a SVG. Kind regards, Marc

mehmetaliincekara (2015-11-09 09:37:31)

Hello Tom, thank you for your response. We will upload a new version of our class diagram with the attributes. We don't see that the mouseover animation over cover the class name. Have a nice day, Ali from nappydevelopment

Spooser | Jani (2015-11-09 09:28:15)

Hello nappydevelopment, -Class diagram that uses correct UML (according to your tool – some tools invent their own version of UML that is close enough) -> Your diagram uses correct UML -Class has to contain attributes and methods. -> Your classes contain attributes and methods -There should be at least five classes. You should be designing classes that are needed for all five use cases promised for Christmas. -> You have more than five classes and I think you have the classes needed to implement your use cases -Not drawn by hand but generated with tool (preferably integrated in IDE for roundtrip engineering) ->You generated your diagram with IntelliJ Your diagrams are very nice, but in the overall diagram is a popup that shouldn't be there. Kind regards, Jan-Eric from saphijaga

marcmahlerdhbw (2015-11-09 09:36:08)

Hello Jani, Thank you very much for your comment. I will remove the popup and reupload the class diagramm as an SVG Kind regards, Marc

Software Architecture Document (2015-11-12 09:00)

Hello everybody :)

you can find our Software Architecture Document (SAD) on GitHub:

[https://github.com/nappydevelopment/docs/blob/master/pdfs/Software %20Architecture %20Document.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Software%20Architecture%20Document.pdf)

We also updated our SRS.

Have a nice day,

Ali from nappydevelopment

@Edit: <https://github.com/nappydevelopment/Nappy-the-ingenious/blob/master/jar/nappy.jar>

(die aktuelle Version ermöglicht das Starten des Spiels und das ändern der Sprache / Spielmodis und Wiki etc. sind deaktiviert in dieser Jar-Version 01)

Spoozzer | Jani (2015-11-16 08:34:55)

Hello nappydevelopment, you have clearly marked the sections of your mvc and showed us your DB model. Are the characters the only thing you store in the database? Do you plan to implement a highscore or something like this? Kind regards Jani from saphijaga

mehmetaliincekara (2015-11-16 08:43:46)

Hello Jani, thank your for reading our Software Architecture Document. We will implement 2 statistics in our statistic-view: most played characters and best parties against Nappy. We think about to separate the database but actually the database contains just information about the characters. Later we will add a second database with the questions Nappy could ask. Have a nice day Ali from nappydevelopment

tom433 (2015-11-16 08:41:14)

Hello nappydevelopment team, you choose JavaFX as MVC framework understandable because you create a Java application. Because of this you don't have a demo version of first use case live on publicly reachable server. Suggestion: You can post a jar file with the first use case so anybody can see your work. Your architectural UML diagram of your application showing clearly marked MVC regions, it's a good idea to seperate the big picture also in single parts :) Your RUP architecture document is filled out and it's online. Kind regards, Tom from tohemidevelopment

mehmetaliincekara (2015-11-16 08:52:25)

Hello Tom, thank you for reading our document. It is a great idea with the jar. You can find out executable jar on GitHub: <https://github.com/nappydevelopment/docs/raw/master/jars/nappy-01.jar> (die aktuelle Version ermöglicht das Starten des Spiels und das ändern der Sprache / Spielmodis und Wiki etc. sind deaktiviert in dieser Jar-Version 01) Have a nice day Ali from nappydevelopment

We are on Jira (2015-11-21 17:13)

Hey :)

we are now on Jira:

<http://193.196.7.27:8080/secure/RapidBoard.jspa?rapidView=8>

We already started our first sprint and we are working on the tasks. This sprint will last 2 weeks.

For example: Some stories (Implement GUI) have subtasks (Implement Mainstage GUI).

Tasks belong to a RUP workflow: [http://sce.uhcl.edu/helm/RationalUnified Process/](http://sce.uhcl.edu/helm/RationalUnifiedProcess/)

Have a nice day

Ali from nappydevelopment

@EDIT: GitHub and our IDE's are connected with JIRA

Spoozer | Jani (2015-11-22 15:02:20)

Hello nappydevelopment, You have a well organized Sprint! It is clearly visible to see who does which work. Well Done! Kind regards, Jani

mehmetaliincekara (2015-11-22 15:05:45)

Hello Jani, thank you for your response and for your praise. Have a nice day, Ali from nappydevelopment

tom433 (2015-11-23 09:09:37)

Hello nappydevelopment, welcome on Jira :) Your first Sprint looks nice and is well organized. Good work ;) Kind regards, Tom from tohemidevelopment

mehmetaliincekara (2015-11-23 09:13:59)

Hello Tom, thank you for your response and for your praise. Have a nice day, Ali from nappydevelopment

Gantt Chart (2015-11-26 12:32)

Hey :)

we uploaded our Gantt Chart with all tasks that we have done until today. We are using MS Project for our Gantt Chart.

https://github.com/nappydevelopment/docs/blob/master/pdfs/SE_GanttChart_UntilMidterm.pdf

https://github.com/nappydevelopment/docs/blob/master/SE_GanttChart.mpp (the .mpp file)

Have a nice day,
Ali from nappydevelopment

tom433 (2015-11-28 13:36:36)

Hello Ali, you have done a good job with the gantt chart. I can see all tasks that you have done until today and the task that you have planed until midterm presentation. They are also seperated according to RUP. The tasks are separated into phases and workflows (disciplines) according to RUP All your Tasks have a date but where can I see the person associated with the tasks? You also don't set a milestone "elaboration" or "midterm persentation" for the second week in December ;) Do you really want to present your homework 10 on 07.12? I think you mean the 09.12 :) Same at the milestone homework 11 on 14.12. . Eventually you could upload a pdf with the gantt chart because not everybody has MS Project on his computer. Kind regards, Tom from tohemidevelopment

mehmetaliincekara (2015-11-28 18:01:31)

Hello Tom thank you for your comment. We wasn't finished with our Gantt-Chart we updated just few minutes after your comment our first final version ;) In the new version there are person associated with the tasks and the milestone for the midterm presentation exists. To create a pdf is a nice idea. We will link it in this post. Have a nice day Ali from nappydevelopment

samuelp94 (2015-11-29 17:21:12)

Hello Team Nappydevelopment, your gantt chart looks great, but in the pdf its very hard to read something directly on github. If you download the file it is ok. Kind regards, Samuel from saphijaga

mehmetaliincekara (2015-11-29 17:24:27)

Hello Samuel, thank you for your comment. We know but you can download the pdf or can click "Bild anzeigen". Have a nice day Ali from nappydevelopment

1.4 December

Midterm (2015-12-18 15:23)

Hallo zusammen :)

das ist unsere kurze Zusammenfassung / Linksammlung für alle Dokumente bis Midterm.

GitHub-Code: <https://github.com/nappydevelopment/Nappy-the-ingenuous>

GitHub-Docs: <https://github.com/nappydevelopment/docs/>

[1]JIRA

[2]Software Requirements Specification

[3]Software Architecture Document

[4]UceCase Gamemode 1

[5]UseCase View Wiki

[6]UseCase View Instructions

[7]UseCase Change Settings (Language)

[8]UseCase View Info

[9]Mockups

[10]Gantt-Chart (PDF)

[11]Gantt-Chart (MPP)

[12]Midterm-PPP

[13]JAR



[14] Euer nappydevelopment-Team

1. <http://193.196.7.27:8080/secure/RapidBoard.jspa?rapidView=8&projectKey=NAP&view=planning>
2. <https://github.com/nappydevelopment/docs/blob/master/pdfs/Software%20Requirements%20Specification.pdf>
3. <https://github.com/nappydevelopment/docs/blob/master/pdfs/Software%20Architecture%20Document.pdf>
4. https://github.com/nappydevelopment/docs/blob/master/pdfs/1%20UseCase_Gamemode%201.pdf
5. https://github.com/nappydevelopment/docs/blob/master/pdfs/2%20UseCase_View_Wiki.pdf
6. https://github.com/nappydevelopment/docs/blob/master/pdfs/3%20UseCase_View_Instructions.pdf
7. https://github.com/nappydevelopment/docs/blob/master/pdfs/4%20UseCase_Change_Settings.pdf
8. https://github.com/nappydevelopment/docs/blob/master/pdfs/5%20UseCase_View_Info.pdf
9. <https://github.com/nappydevelopment/docs/blob/master/pdfs/Mockups.pdf>
10. https://github.com/nappydevelopment/docs/blob/master/pdfs/SE_GanttChart_UntilMidterm.pdf
11. https://github.com/nappydevelopment/docs/blob/master/SE_GanttChart.mpp
12. <https://github.com/nappydevelopment/docs/blob/master/Midterm-PPP.pptx>
13. <https://github.com/nappydevelopment/Nappy-the-ingenuous/blob/master/jar/Nappy-the-ingenuous.jar>
14. <https://github.com/nappydevelopment/docs/blob/master/pdfs/Mockups.pdf>

2. 2016

2.1 April

Riskmanagement and planned UseCase's (2016-04-10 18:00)

Hey guys,

for this semester we updated our OUCD (Overall UseCase Diagramm):

[https://github.com/nappydevelopment/docs/blob/master/pdfs/Overall %20Use %20Case %20Diagram.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Overall%20Use%20Case%20Diagram.pdf)

We are planning to implement 4 further UseCases:

UseCase 6: Gamemode 2 - [https://github.com/nappydevelopment/docs/blob/master/pdfs/6 %20UseCase _Gamem-ode %202.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/6%20UseCase_Gamemode%202.pdf)

UseCase 7: Extend Wiki - [https://github.com/nappydevelopment/docs/blob/master/pdfs/7 %20UseCase _Extend %20Wiki.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/7%20UseCase_Extend%20Wiki.pdf)

UseCase 8: View Statistic - [https://github.com/nappydevelopment/docs /blob/master/pdfs/8 %20UseCase _View _Statistic.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/8%20UseCase_View_Statistic.pdf)

UseCase 9: Extend Settings - [https://github.com/nappydevelopment/docs /blob/master/pdfs/9 %20UseCase _Extend _Settings.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/9%20UseCase_Extend_Settings.pdf)

UseCase Time-effort: [https://github.com/nappydevelopment/docs/blob/master/pdfs/Usecase %20doc.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Usecase%20doc.pdf)

Also you can find our riskmanagement on GitHub: <https://github.com/nappydevelopment/docs/blob/master/pdfs/Riskmanagment.pdf>

Have a nice day

Ali from nappydevelopment

tom433 (2016-04-11 07:29:19)

Hello Ali, nice work ;) I hope I can try the new gamemode asap. :) All of your documents look great but one or more use cases are wrote in english and others in german. Eventually it would be better if you wrote all documents in one language :) Kind regards, Tom from tohemidevelopment

mehmetaliincekara (2016-04-11 07:48:34)

Hello Tom, first thank you for your comment. Yeah, this is right <(o.o<) we will change it. Have a nice day, Ali from nappydevelopment

Spoozer | Jani (2016-04-11 07:38:19)

Hello nappy, You have done nice work, but there are still some things I want to say: -Some of your documents are english and some are german which is not nice -Why do you have a revision control in your document? Its useless when you are using GitHub -The Riskmanagement is not marked as link -The link for the use case time effort is the link for risk management Greetings, Jani from Spoozer

mehmetaliincekara (2016-04-11 07:46:58)

First of all thank you for your response. Yeah, this is right <(o.o<) we will change it. We have the revision control that the person who want to see what happend in the document don't have to look at GitHub. We linked our riskmanagement and

UC-Time table. Have a nice day, Ali from nappydevelopment

Function points (2016-04-17 17:00)

Hey guys,

we just finished our calculation of our function points:

Our result: [https://github.com/nappydevelopment/docs/blob/master/pdfs/Usecase %20doc.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Usecase%20doc.pdf)

Outliers explanation are in the document.

Have a nice day

Ali from nappydevelopment

tom433 (2016-04-18 06:27:43)

Hello Ali, good work :) Eventually you can explain the outliers in the diagram. ;) Kind regards, Tom from tohemidevelopment

mehmetaliincekara (2016-04-18 06:43:25)

Hey, thank you for your comment. We updated our post with explanation's. Have a nice day Ali from nappydevelopment

MAy mustermann (2016-04-18 06:34:42)

Hey guys, First of all your use cases are very well documented. Also you did the FP calculation for your old and new use cases. Furthermore you included the calculation into your UC document. The graph shows the time spent per FP for your 5 implemented use cases. But it would be nice if you could explain why the first dot is way below your line. Additionally you estimated the time for your new use cases by using the graph. Well done :) Kind Regards, Team Unveiled

mehmetaliincekara (2016-04-18 06:43:12)

Hey, thank you for your comment. We updated our post with explanation's. Have a nice day Ali from nappydevelopment

Daniel Brown (2016-04-25 09:02:34)

Hi NappyDevs, your estimation looks good and your estimation curve is pretty accurate. Did you calculate the weight with DET and FTR or did you just estimate it? Regards, BooksterDan

mehmetaliincekara (2016-04-25 09:24:19)

Hi, thank you for your comment. We just estimate the weight.

TDD, JUnit and UC's (2016-04-23 13:37)

Hey guys,

first we will talk about our planned Use Case's and how we decided which one we will implement.

At the beginning of our project we thought about "what function do we want?"

Our result was this excel-sheet: [https://github.com/nappydevelopment/docs/blob/master/pdfs/planned %20UC.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/planned%20UC.pdf)

At the beginning of our second semester we thought about which of the remaining functions we can implement. The result is also in this excel-sheet.

For the next part of our homework we did a screencast to show how we programmed a simple calculator with TDD. You can find the video in our documentation repository: [https://github.com/nappydevelopment/docs-
/blob/master/other %20homework/Screencast %20for %20TDD.mp4](https://github.com/nappydevelopment/docs/blob/master/other%20homework/Screencast%20for%20TDD.mp4)

We also implemented tests for at least one class of our project. For prove, you can look at the following picture: [https://github.com/nappydevelopment/docs /blob/master/other %20homework/OneClassTested.png](https://github.com/nappydevelopment/docs/blob/master/other%20homework/OneClassTested.png)

Our test code is provided in github too, so if you like you can watch our full test code in our project.

We also have integrated the tests into our maven build process, so all tests run at the build process and we see immediately what is wrong with our code. Actually the code is implemented in a branch: <https://github.com/nappydevelopment/Nappy-the-ingenious/tree/maven>

tom433 (2016-04-25 07:36:45)

Hello Ali, first of all good work ;) Your decision with excel is clearly defined and everyone can see why you do which use case. The video of the TDD part and the Junit tests are also fine. Eventually you can add a link to the file in github so others have not to search for it. Kind regards, Tom from tohemidevelopment

mehmetaliincekara (2016-04-25 07:39:37)

Hello Tom, thank you for your comment. The link to maven is already added. Have a nice day Ali from nappydevelopment

samuelp94 (2016-04-25 07:54:09)

Hello nappydevelopment, your use case planning and your test code looks nice. Your TDD example could be a bit better, e.g. you have not tested the divide by zero error and in most cases one test per method is not enough to test the whole functionality. Kind regards, Samuel from saphijaga

mehmetaliincekara (2016-04-25 08:25:50)

Hello, thank you for your comment. This is right but we wont update the code because it is only toycode and the only purpose was to demonstrate tdd but thank you. Have a nice day Ali from nappydevelopment

2.2 May

Fowler's Refactoring (2016-05-01 11:45)

Hey guys,

we are proud to present you our results of this weeks homework. Each of our team members worked through Chapter one of the "Refactoring: Improving the Design of Existing Code" book by Martin Fowler. We used [1]this github repo as the base. Here are our results:

- [2]Ali
- [3]Marc
- [4]Marvin
- [5]Manuel

Enjoy the long weekend!
Marvin from nappydevelopment

1. <https://github.com/gnilkreb/Fowler/tree/53732ac7430be4a1e04d41a1eb8f68d1af138253>
2. https://github.com/xnappy/Fowler_Refactor
3. <https://github.com/Uindemia/Fowler-Refactor>
4. <https://github.com/madblobfish/Fowlers-Refactor>
5. https://github.com/ManuelBothner/SE_Fowler-Refactoring

Spoozzer | Jani (2016-05-02 07:19:49)

Hello, your GitHub repositories look good. You made a good job in refactoring and committed all steps so everyone can comprehend your progress. Kind regards, Spoozzer

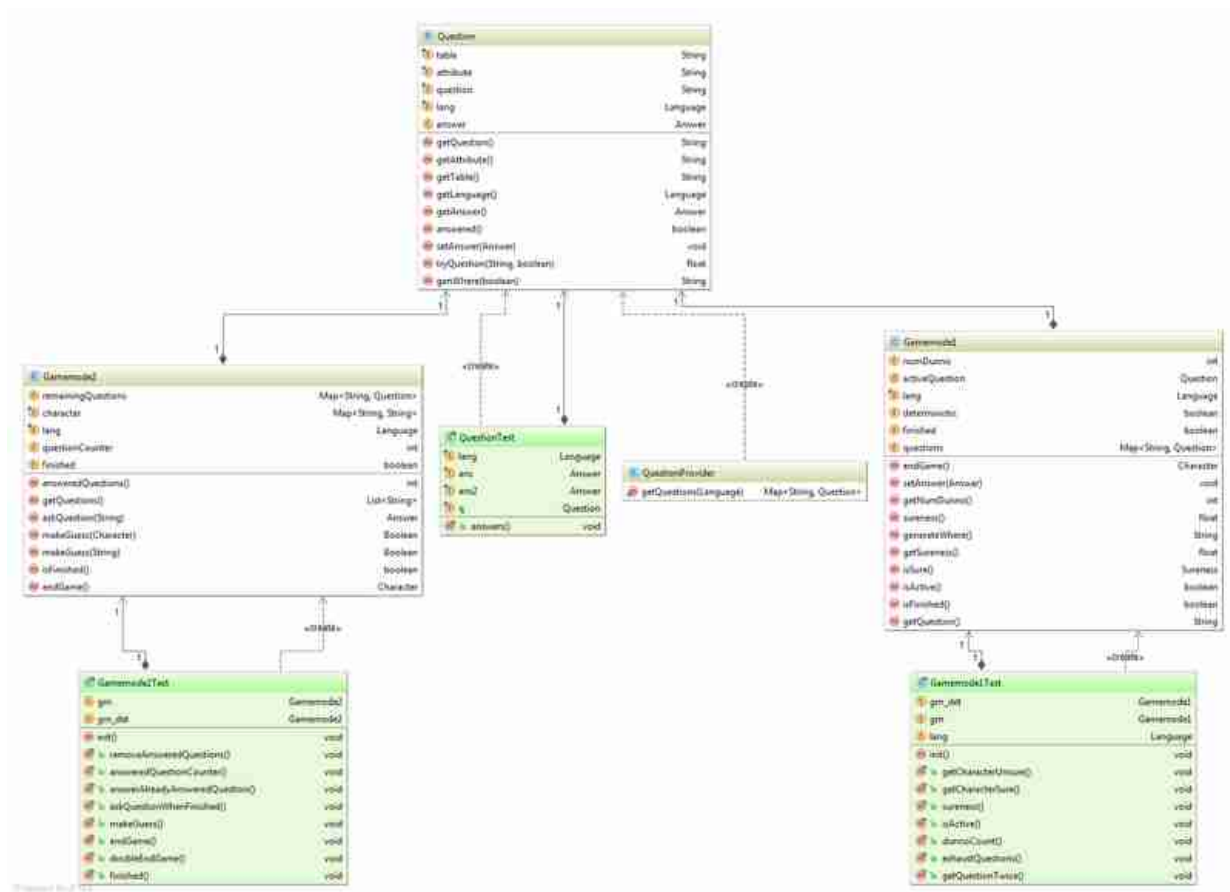
tom433 (2016-05-02 08:11:33)

Hello, you have done good work. All of your teammates have done the work and commit every step by Fowler to GitHub so everyone can see the refactoring progress. Kind regards, Tom from tohemidevelopment

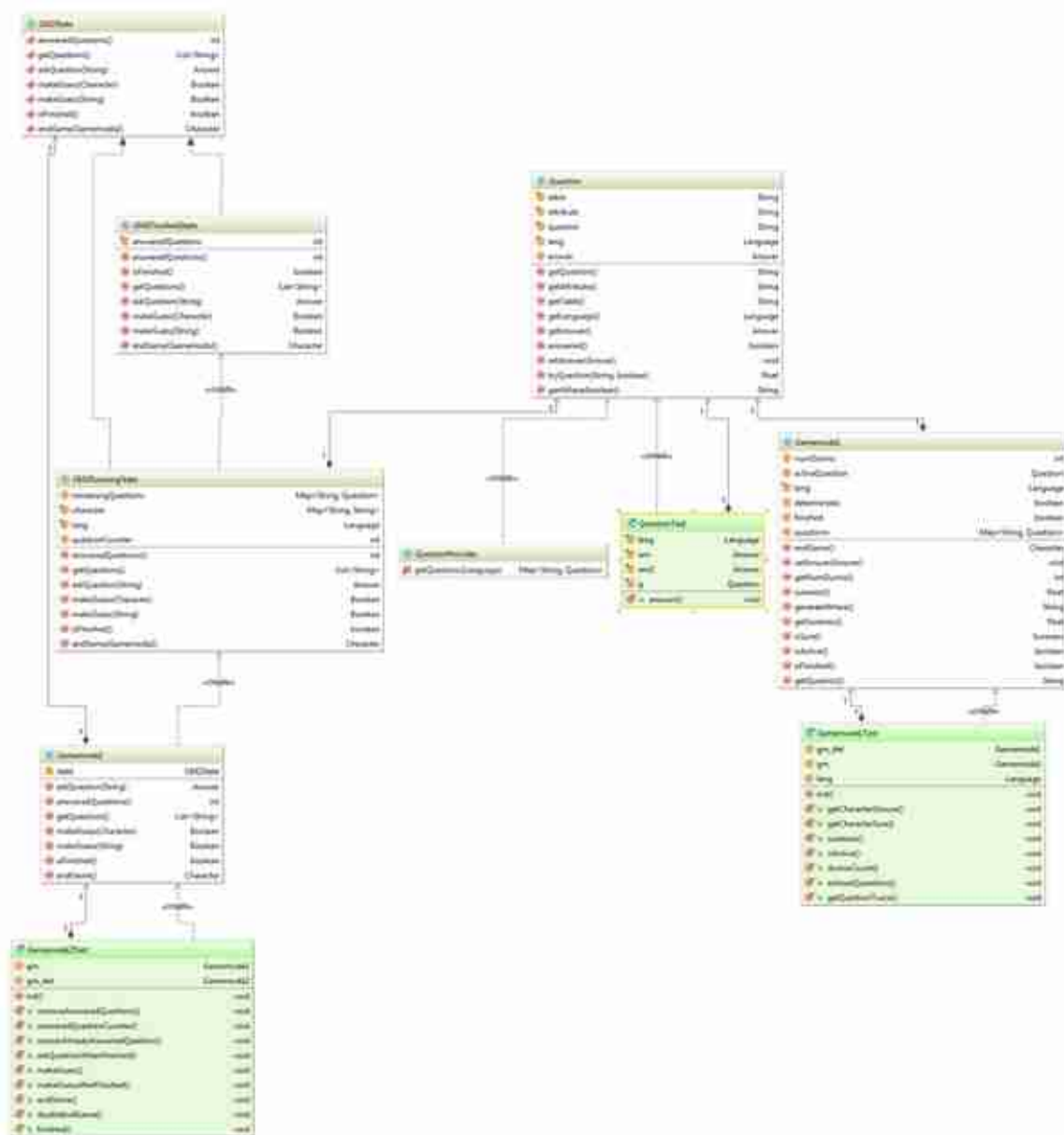
Design Pattern (2016-05-09 09:13)

We are using the State pattern for our Gamemode2 class. It helped us eliminate one instance variable which represented the state.

UML ([1]Code) before:



UML ([2]Code) after:



Greetings from nappydevelopment

Edit (2016-05-19):

We also moved GameMode1 to Statepattern
[3]before, [4]after.

1. https://github.com/nappydevelopment/Nappy-the-ingenuous/tree/59f11b1b2d893fec917478640996b4b90f1c223/src/main/java/nappydevelopment/nappy_the_ingenuous/gamemodes/gamemode2
2. <https://github.com/nappydevelopment/Nappy-the-ingenuous/tree/e9962241780330a8ca88903c090e6090c8bae596/src/>

main/java/nappydevelopment/nappy_the_ingenious/gamemodes/gamemode2

3. <https://github.com/nappydevelopment/Nappy-the-ingenious/tree/30d7a845ade17e754cb768ca43e4054f5fa80f4e/src/main/java/nappydevelopment/nappyTheIngenious/gamemodes/gamemode1>

4. <https://github.com/nappydevelopment/Nappy-the-ingenious/tree/e2ee4dfaf2d705bb80bca424fbc230a9ab28275/src/main/java/nappydevelopment/nappyTheIngenious/gamemodes/gamemode1>

Spoozer | Jani (2016-05-09 13:07:16)

Hello, you have done good work implementing the pattern, but you could add the pictures directly in the blog entry so I don't have to click on the links to see it. I think your links to "Code Before" and "Code After" are wrong and why do you have in the "Code After" a method "isFinished"? Kind regards, Jani from Spoozer

mehmetaliincekara (2016-05-09 13:13:12)

Hello, the pictures are included directly in the blog entry. We will change "Code Before" and "CodeAfter". Have a nice day, Ali from nappydevelopment

marvinzerulla (2016-05-09 13:18:50)

Hello, I don't see anything wrong with the isFinished() method. Was present in the before version and belongs to the API. It is used to determine the state from the outside. Greetings Marvin from nappydevelopment

tom433 (2016-05-09 13:14:24)

Hello, you have implemented the state pattern well. But I think you must change the links to the github repository because you exchange them. Also fine job that you now have the pictures in the blogpost. Kind regards, Tom from tohemidevelopment

mehmetaliincekara (2016-05-09 13:15:51)

Hello, thank you for your comment. We updated the links. Have a nice day, Ali from nappydevelopment

Metrics (2016-05-29 17:00)

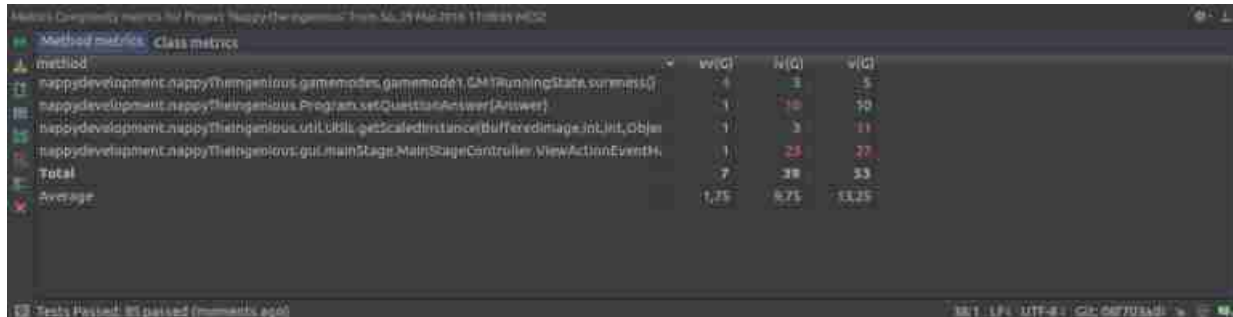
Hey guys,

we are using the metrics plug-in for IntelliJ: MetricsReloaded

We are using following metrics profiles: Complexity and Dependency

metrics before:

[1]



The screenshot shows the 'Method metrics' tab in the IntelliJ MetricsReloaded plugin. It displays a table with three columns: $w(G)$, $N(G)$, and $v(G)$. The data is as follows:

Method	$w(G)$	$N(G)$	$v(G)$
nappydevelopment.nappythingenious.gamemodes.gamemodet.CMRunningState.summestG	1	3	5
nappydevelopment.nappythingenious.Program.setQuestionAnswer[Answer]	1	10	10
nappydevelopment.nappythingenious.util.UBLL.getScaledInstance(BufferedImage,int,int,Object)	1	3	11
nappydevelopment.nappythingenious.gui.mainStage.MainStageController.ViewActionEventH	1	23	27
Total	7	39	53
Average	1.75	9.75	13.25

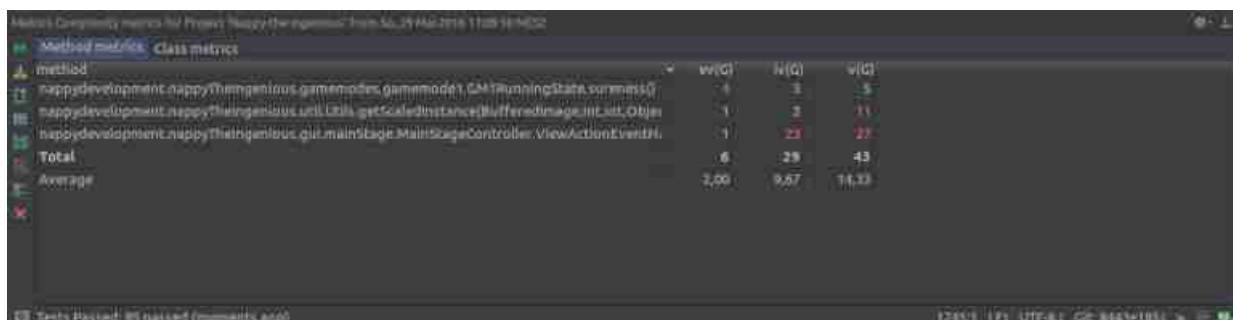
$ev(G)$ Calculates the essential complexity of each non-abstract method. Essential complexity is a graph-theoretic measure of just how ill-structured a method's control flow is.

$iv(G)$ Calculates the design complexity of a method. The design complexity is related to how interlinked a method's control flow is with calls to other methods.

$v(G)$ Calculates the cyclomatic complexity of each non-abstract method. Cyclomatic complexity is a measure of the number of distinct execution paths through each method. In practice, this is 1 + the number of if's etc. in the method.

metrics after: (only the worst are shown, the method had following scores after: 2, 4, 7)

[2]



The screenshot shows the 'Method metrics' tab in the IntelliJ MetricsReloaded plugin after refactoring. It displays a table with three columns: $w(G)$, $N(G)$, and $v(G)$. The data is as follows:

Method	$w(G)$	$N(G)$	$v(G)$
nappydevelopment.nappythingenious.gamemodes.gamemodet.CMRunningState.summestG	1	3	5
nappydevelopment.nappythingenious.util.UBLL.getScaledInstance(BufferedImage,int,int,Object)	1	3	11
nappydevelopment.nappythingenious.gui.mainStage.MainStageController.ViewActionEventH	1	23	27
Total	6	29	43
Average	2.00	9.67	14.33

The method we tried to fix was `Program.setQuestionAnswer` ([3]code before) ([4]code after). This was rather simple as this method has grown lately, it really only needed one try-catch. Also the counting of don't knows was moved into `gamemode1`.

Another spot in our code where metrics suggest that should change is our method `"ViewActionEventHandler"` in our [5]`MainStageController` but we have decided not to. There are so many if's because if someone starts the game the `MainStage` will "changed" to the `GameStage`. We don't have an extra stage for both game modes because they are part of the `MainStage` and the transition is better. We can't and won't change this method because we don't have enough time.

The tool isn't part of our deployment process because it wasn't possible. We have Codacy, Coveralls and Sonarqube in our deployment process to check our code quality. But we thought we should present `"MetricsReloaded"` because it is a really powerfull tool.

Ali and Marvin
from nappydevelopment

1. <https://nappydevelopment.files.wordpress.com/2016/05/metriks-vorher2.png>
2. <https://nappydevelopment.files.wordpress.com/2016/05/metriks-nachher2.png>
3. <https://github.com/nappydevelopment/Nappy-the-ingenuous/blob/8443e185f496912db35a1d89b1ed29c2248ca1f8%5E/src/main/java/nappydevelopment/nappyTheIngenious/Program.java#L282>
4. <https://github.com/nappydevelopment/Nappy-the-ingenuous/blob/8443e185f496912db35a1d89b1ed29c2248ca1f8/src/main/java/nappydevelopment/nappyTheIngenious/Program.java#L282>
5. <https://github.com/nappydevelopment/Nappy-the-ingenuous/blob/master/src/main/java/nappydevelopment/nappyTheIngenious/gui/mainStage/MainStageController.java>

tom433 (2016-05-30 06:50:17)

Hello Ali and Marvin, you use a metrics tool. Is this tool a part of your deployment process? If yes can you tell us where you start it. You have refactor a code snippet and I can see the result of it. You also explain why this is a good point to refactor it. Also you give an example of code where you don't refactor your code. Kind regards, Tom from tohemidevelopment

mehmetaliincekara (2016-05-30 06:55:43)

Hello Tom, thank you for your comment. The tool isn't part of our deployment process because it wasn't possible. We have Codacy, Coveralls and Sonarqube in our deployment process. But we thought we should present `"MetricsReloaded"` because it is a really powerfull tool. Have a nice day Ali from nappydevelopment

Spoozier | Jani (2016-05-30 09:00:17)

Hello nappydevelopment, you showed us which metrics tool you are using, you explained why it is not part of your deployment process, you refactored your code where the tool showed that you have a problem and also gave us an example where you won't change it and explained why. You did a nice job! Have a nice week, Spoозer

mehmetaliincekara (2016-05-30 09:26:13)

Hello Jani, thank you for your comment. Have a nice day Ali from nappydevelopment

Automatic deploy and Test Plan (2016-05-30 11:30)

Hey guys,

here is the link to our Test Plan:

[https://github.com/nappydevelopment/docs/blob/master/pdfs/Test %20Plan.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Test%20Plan.pdf)

In the testplan are our coverage tools, test types, etc. declared.

In our SAD is our deployment process.

[https://github.com/nappydevelopment/docs/blob/master/pdfs/Software %20Architecture %20Document.pdf](https://github.com/nappydevelopment/docs/blob/master/pdfs/Software%20Architecture%20Document.pdf)

Have a nice day,

Ali from nappydevelopment

Spooser | Jani (2016-06-03 08:36:34)

Hello, your Test plan looks nice, but your informations about the deployment are missing.Maybe you could also link your SAD directly to this post Kind regards, Spooser

mehmetaliincekara (2016-06-03 09:42:54)

Hello Jani, we forgot to push the updates for our SAD. Thank you for your comment. Have a nice day Ali from nappydevelopment

tom433 (2016-06-03 08:56:28)

Hello Ali, your Test plan looks fine, but the information about the deployment process are missing. Kind regards, Tom from tohemidevelpment

mehmetaliincekara (2016-06-03 09:43:11)

Hello Tom, we forgot to push the updates for our SAD. Thank you for your comment. Have a nice day Ali from nappydevelopment



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

Edited: June 6, 2016

