

p

~ Veröffentlicht am 13. Oktober 2016 ~

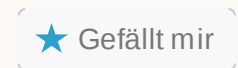
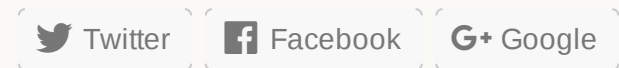
## About us

---

**D**URING OUR SOFTWARE ENGINEERING LECTURE IN THE third and fourth semester we, a group consisting of three students (Sandra, Sven, Dominik), have to develop an application with our new achieved skills.

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

Software Requirements  
Specification  
In "Allgemein"

DB-Schema & Class  
diagram  
In "Allgemein"

Jira  
In "Allgemein"

NEXT ARTICLE

*The Vision*

## EIN GEDANKE ZU “ABOUT US”

*Pingback: Final Presentation – PoS Systems*

### *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Folgen



p

~ Veröffentlicht am 13. Oktober 2016 ~

## The Vision

---

**W** E ARE PLANNING TO DEVELOP A PRODUCT GROUP BASED cash application for retailers with a small system to manage product groups and employees and a particular area for statistics.

Our ambition is to create an application, which fulfills all legal requirements and still is user-friendly. It should deliver all functions needed in a small retail. Therefore the user needs the possibility to create and manage product

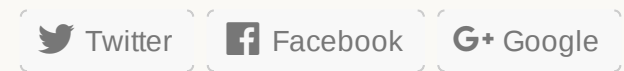
groups and employee numbers, which are used as identifier in the system. There should also be the possibility to create some statistics with the old selling processes. During a selling process the user can select a product group and the price. After all products are added the user selects the payment methods and finishes the sales process. Different roles decide the access rights for employees. Depending on the rights, a user will be directly redirected to the cash or can decide rather he wants to use the cash or the management system.

The whole project will be divided into some smaller parts. First we want to create a simply cash layout, followed by the logic for the main sales process (e.g. adding some products with price and save them in the database). After that we will expand it by features like sale discount and cancellation. Other necessary parts will be the cash opening, cash check and the cash account closing. After the cash is working, we will go over to the management system, which will be divided by its features.

Depending on our possibility to get hardware like a receipt printer and a cash box, we also try to create a connection to these components.

Advertisements

Teilen mit:



Lade ...

Ähnliche Beiträge

**Code Pattern**  
In "Allgemein"

**Function Points**  
In "Allgemein"

**Final Presentation**  
In "Allgemein"

PREVIOUS ARTICLE

*About us*

NEXT ARTICLE

*Technologies and Team Roles*

## 5 GEDANKEN ZU “THE VISION”



SMARTCINEMAPROJECT *sagt:*

13. OKTOBER 2016 UM 14:13

Sounds like an interesting project. Your explanation is short and understandable and the objective of your project is clear. How about any subsystems? Can you break your project down in smaller pieces?

★ Gefällt 1 Person





POS SYSTEMS *sagt:*

13. OKTOBER 2016 UM 16:32

It's hard to split it up, because there are a lot of features which are definitely needed and can't be left out, but of course we try it.

We differ between the management system, which doesn't have many features and the more extensive cash system. The management system will be one part itself, but the cash will be divided into smaller steps. First we want to create a simple layout, followed by the logic for the main sales process (e.g. adding some products with price and save them in the database). After that, we'll expand it by features like sale discount and cancellation. Other necessary parts will be the loan, cash check and the cash balance.

★ Gefällt mir

*Antwort*



TECHNIKO27 *sagt:*

17. OKTOBER 2016 UM 10:52

Hey,  
your project is very useful!

I like your detailed description through the work flow.

Do you also think about to add functions to analyse data, like which product is most sold?

Do you also implement the interface to a cash card terminal or a sales slip printer?

★ Gefällt mir

*Antwort*



POS SYSTEMS *sagt:*

17. OKTOBER 2016 UM 16:53

Hey, thank you for your feedback.

We thought about a particular area in the management system with some statistics and the possibility to have a look at the old selling processes. But the implementation depends on the time left because it's not an essential feature. We also have to mention that it's just a cash for product groups, therefore it's not possible to create statistics about specific articles or their EAN.

Our main objection is to create a working cash which also includes the connection to a receipt printer and a cash box, but not a cash card terminal. But right now we don't know how to get the relevant hardware for test purposes without buying them.

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 15. Oktober 2016 ~

## Technologies and Team Roles

---

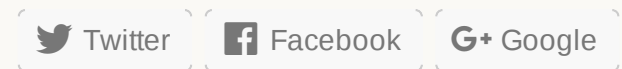
**O**UR APPLICATION IS GOING TO BE REALISED AS A JAVA web application based on the Spring framework. In the background it will be connected to a MySQL database to store the master data and the data resulting from the selling processes. The IDE of choice will be eclipse with the Spring Toolsuite (STS) plugin, as this is the recommendation of the website „spring.io“ and there are a lot of tutorials about the spring framework using this IDE.

These are our roles:

Designer	Sven Baumann
Requirements Specifier	All
Implementer	All
Deployment	Sandra Kramlich
Tester	Dominik Schneider
Project Manager	Sandra Kramlich
Tool Specialist	Dominik Schneider
Configuration and Change Manager	Sven Baumann

Advertisements

Teilen mit:



Lade ...

### Ähnliche Beiträge

**Refactoring**  
In "Allgemein"

**Final Presentation**  
In "Allgemein"

**The Vision**  
In "Allgemein"

PREVIOUS ARTICLE

*The Vision*

NEXT ARTICLE

*Software Requirements Specification*

## 4 GEDANKEN ZU “TECHNOLOGIES AND TEAM ROLES”



TINF15B4LSMF *sagt:*

16. OKTOBER 2016 UM 12:05

Hi team,

your role distribution seems pretty solid, all important roles are assigned. The thing you haven't written about is the IDE you are going to use, which would be interesting to know. We are also curious where going to host the finished product.

★ Gefällt 1 Person

*Antwort*



POS SYSTEMS *sagt:*

16. OKTOBER 2016 UM 16:07

Hey, thanks for your comment. Right now we don't know exactly which IDE we will use, but it will be either Eclipse or the Spring Tool Suite (STS). It's also unclear

where exactly we will host the product, but we have some tendencies, which will get more precise in the following days.

★ Gefällt mir

*Antwort*



MUMSFIT *sagt:*

17. OKTOBER 2016 UM 10:35

Hello from MuM,

good job in showing your different roles throughout the project using the correct RUP terminology. Also, you did not forget to mention that you will use a database to store the information. Good job!

Kind regards, MuM

★ Gefällt 1 Person

*Antwort*

*Pingback: Final Presentation – PoS Systems*



## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 24. Oktober 2016 ~

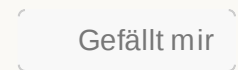
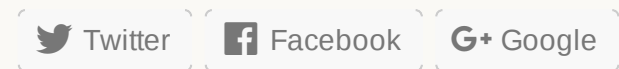
## Software Requirements Specification

---

**T**HE FOLLOWING LINKS WILL REDIRECT YOU TO OUR  
**Software Requirements Specification** and our **Use-Case-Diagramm**.

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

Software Architecture  
Document  
In "Allgemein"

DB-Schema & Class  
diagram  
In "Allgemein"

First two Use-Cases  
In "Allgemein"

PREVIOUS ARTICLE

*Technologies and Team Roles*

## 5 GEDANKEN ZU “SOFTWARE REQUIREMENTS SPECIFICATION”



TINF15B4LSMF *sagt:*

25. OKTOBER 2016 UM 20:17

Hey there,

There SRS is looking good so far, but perhaps you should maintain the blue fields and add some kind of „tbd“ or „n/a“, what you have so far looks to be correct, though!  
Keep it up.

Now, as for the use-case-diagramm, the link you provided is linking to the SRS which doesn't include the diagram – at least if viewed in libre office. Couldn't find it in the github branches either.

Apart from the grading criteria, there's some personal questions that arose from the SRS that shouldn't relate to the lecture at this point in time:

Do you think that 15 minutes to one hour as a mean time to repair is a realistic goal? If so, the customer have to file a report like that with you directly or are you going to implement a „call support“ function of some kind?

Also, how long do you think you'll be able to persist the data amassed by your systems? With a sensitive topic such as finances that potentially generate a lot of data, persistence becomes quite tricky. Will you treat that as out of scope for the lecture for now or are you planning on taking any measures in regards to that?

Take care and keep up the good work,

Team Silver to the Screen~

★ Gefällt 1 Person

*Antwort*



POS SYSTEMS *sagt:*

26. OKTOBER 2016 UM 20:35

Thanks for your feedback. We know that the mean time to repair is quite short and we reconsidered it but the reason for the short time are our users. Our users are retailers which rely on our application. Every minute of a stopped cash means financial loss for them, so minor bugs or problems should be fixed as fast as

possible (<30 minutes) or should be avoided through hotfixes. Therefore a telephone hotline would be the ideal solution as contacting method to minimize the downtime.

We mentioned that our application should fulfill all legal requirements, this includes the data preservation for 10 years and to be picky 13 years. Something like a RAID System as protection against data loss is out of scope for the lecture.

★ Gefällt mir

*Antwort*



SMARTCINEMAPROJECT *sagt:*

26. OKTOBER 2016 UM 13:48

Your use case diagram looks a little crowded. Maybe you can group up some use cases. e.g. „use-cash-area“ includes several other use cases. maybe you could create a „use-cash-area“-group and move the other use cases into it. Another point is that not all use cases contain a verb so they are not ideal. Maybe you should check that. Apart from that the use case diagram looks very detailed and complete.

In your SRS you should replace the blue lines either with your own text, or if you don't know what it is about with n/a (not available) or tbd (to be determined). Your SRS contains all necessary points and the versioning control. All points seem to be correct/realistic, but the repair time of 15 to 60 minutes in case of data loss seems very optimistic. The other way around, one transaction per seconds seems very slow for a financial system. You might want to think about those again.

Kind regards  
Team of SmartCinema

★ Gefällt 1 Person

*Antwort*



POS SYSTEMS *sagt:*

26. OKTOBER 2016 UM 20:54

As I just answered above we reworked our mean time to repair. The repair time also relates to corrupt data which should be possible to recreate in a specific time (<2 hours) in which this isn't allowed by law. Data loss is something more problematic but should be prevented through the application itself.

One transaction per second seems slow but depends on how you define the transaction. In our case it includes the whole process after pressing the "paid" button until you're back in the main window to start a new selling process. Therefore the maximum of a second should be realistic.

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen



p

~ Veröffentlicht am 1. November 2016 ~

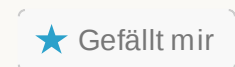
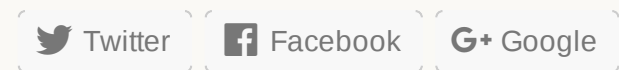
## First two Use-Cases

**T**HE FOLLOWING LINKS WILL LEAD YOU TO OUR FIRST TWO  
Use-Cases:

1. [Manage Product Groups](#)
2. [Select between Management- & Cash-Area](#)

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

**Software Requirements  
Specification**  
In "Allgemein"

**.feature**  
In "Allgemein"

**Semester Start**  
In "Allgemein"

PREVIOUS ARTICLE

*Software Requirements Specification*

NEXT ARTICLE

*Jira*

## 4 GEDANKEN ZU “FIRST TWO USE-CASES”



MARCO KOLB *sagt:*

2. NOVEMBER 2016 UM 11:06

Hi,

I had a look at your use-case specification and I wasn't able to find any mock-ups... It would be very helpful to get an overview how your application will look like and to understand your flow diagram.

Moreover, I would define user action like „klicks ...“ as a own flow bubble, beacuse it's easier to understand.

Keep up your work!

Best regards,  
Marco Kolb

★ Gefällt mir

*Antwort*



SMARTCINEMAPROJECT *sagt:*

2. NOVEMBER 2016 UM 14:40

Your use-case specifications utilize the required templates, and they are very clear and understandable and appear to be properly linked to from the SRS.

I had some trouble wrapping my mind around the point in the „manage product groups“ activity diagram where the lines from the decision node and the line from the starting point merge towards the „Abbrechen“ end. Maybe this could be realized using a join node, or you could just drop the direct connection from start to finish because I don't think it conveys anything meaningful (or leave it as is-maybe I'm just the one dumb guy who didn't get it).

By the way, the „manage product groups“ use case looks like it actually is a CRUD. If this is indeed the case, you should explicitly call that out in the use-case specification.

★ Gefällt mir

*Antwort*



POS SYSTEMS *sagt:*

2. NOVEMBER 2016 UM 15:45

Thank you for your feedback. We fixed everything you mentioned except for the mockup. We will make this as soon as possible.

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...



p

~ Veröffentlicht am 9. November 2016 ~

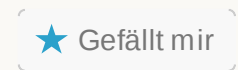
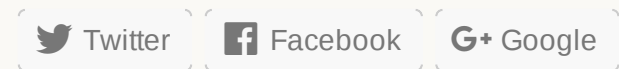
## Jira

O

UR PROJECT IS NOW AVAIABLE ON **JIRA**.

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

Testing 2  
In "Allgemein"

.feature  
In "Allgemein"

Gantt-Diagramm  
In "Allgemein"

PREVIOUS ARTICLE

*First two Use-Cases*

NEXT ARTICLE

*.feature*



### 3 GEDANKEN ZU “JIRA”



4ND3R50N *sagt:*

10. NOVEMBER 2016 UM 10:02

Hello possystemsweb,

Your scrum boards seems rightly configures, but its a little bit empty. But you already said in the lession, that you guys wont really focus on jira []

Well, my suggestion is to add a little bit more cases. Use stories like „develop system X“ and add all tasks which are necessary to complete this story. For 3 people it shouldnt be so much efford.

Keep it up!

Sincerely,  
WhiteCode

★ Gefällt mir

*Antwort*



DIMITAR DIMITROV *sagt:*

10. NOVEMBER 2016 UM 13:06

Hello,

it seems your Jira is configured but you should put some issues and the sprint to cover some of the grading criteria

-Project Centaur

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

*Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 16. November 2016 ~

## .feature

---

O

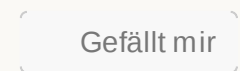
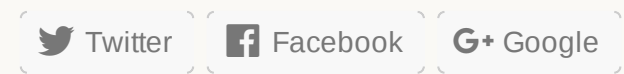
UR FIRST TWO FEATURE FILES ARE AVAIABLE NOW:

1. **Select between Management- & Cash-Area**
2. **Manage Product Groups**

For proof of autocompletion, please click **here**.

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

PREVIOUS ARTICLE

*Jira*

NEXT ARTICLE

*DB-Schema & Class diagram*

## 3 GEDANKEN ZU “.FEATURE”



SMARTCINEMAPROJECT *sagt:*  
16. NOVEMBER 2016 UM 19:56

Hey guys,  
we took a look at your .feature-files and they seem to be in legit cucumber spelling.  
Also they reproduce the flow your described in your use case documents.

The auto completion/syntax highlighting works, too.

Unfortunately, the links in your use case documents aren't clickable. But it shouldn't

be that hard to fix this, right?! []

Keep up your good work!

SmartCinemaProject

★ Gefällt mir

*Antwort*



MUMSFIT *sagt:*

17. NOVEMBER 2016 UM 9:45

Hello from MuM,

your .feature files look good and it looks like they use a legit syntax. Also, you have multiple scenarios for each usecase. The proof of autocompletion also fine, good job!

Kind regards, MuM.

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

*Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen



p

~ Veröffentlicht am 21. November 2016 ~

## DB-Schema & Class diagram

**T** HIS WEEK WE CREATED OUR **DB SCHEMA** AND OUR **CLASS diagram**.

Both were designed with MS Visio 2010.

Advertisements

Teilen mit:



Lade ...

### Ähnliche Beiträge

Software Requirements  
Specification  
In "Allgemein"

First two Use-Cases  
In "Allgemein"

Gantt-Diagram  
In "Allgemein"

PREVIOUS ARTICLE

*.feature*

NEXT ARTICLE

## 4 GEDANKEN ZU “DB-SCHEMA & CLASS DIAGRAM”



SMARTCINEMAPROJECT *sagt:*  
22. NOVEMBER 2016 UM 20:36

Hi POS Team,

your database schema looks good but we found a few points about your class diagram. At first we needed to include at least five classes. I admit that you did exactly five but you should create classes for all five use cases that you promised to do until christmas. I'm not sure if you can cover five use cases with those classes.

As a second point we are missing some kind of proof that you did generate the UML instead of drawing it by hand, for example you could add a screenshot.

Apart from those points the diagram contains attributes and methods and uses correct syntax.

Kind regards!

## Team of SmartCinema

★ Gefällt mir

*Antwort*



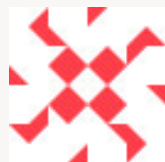
POS SYSTEMS *sagt:*

24. NOVEMBER 2016 UM 11:42

For the UC we've defined to do until christmas the shown calsses are all we need.  
More classes will follow when we start to implement the management area.

★ Gefällt mir

*Antwort*



GAMESCOBOLD *sagt:*

23. NOVEMBER 2016 UM 14:40

Hi,  
it is nice that you have a DB-Schema.  
you have enough classes, but maybe you will later notice that you need more of them.  
In summary it looks okay in comparison with the Grading Criteria.  
Kind regards.

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 24. November 2016 ~

## Software Architecture Document

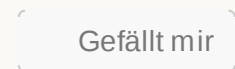
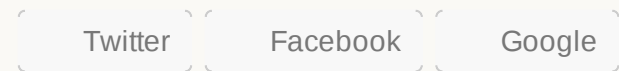
---

**Y**OU CAN FIND OUR SOFTWARE ARCHITECTURE DOCUMENT  
**here.**

Right know we're having some trouble to autogenerate the classdiagram but it will follow.

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

**Software Requirements  
Specification**  
In "Allgemein"

**Testing 2**  
In "Allgemein"

**About us**  
In "Allgemein"

PREVIOUS ARTICLE

*DB-Schema & Class diagram*

NEXT ARTICLE  
*Gantt-Diagram*

## 4 GEDANKEN ZU “SOFTWARE ARCHITECTURE DOCUMENT”



TINF15B4LSMF *sagt:*

28. NOVEMBER 2016 UM 10:19

Hey guys,

according to our grading criterias there were some additional fields to fill in our SAD. Section 2,3,5,7,9 had to to be filled accordingly.

As far as i have noticed, point 3 is missing the MVC Tool you are using, as well as point 7 is missing your DB diagram (UML) in case you use a DB.

Everything else seems to be in order.



Regards  
LSMF

★ Gefällt mir

*Antwort*



SMARTCINEMAPROJECT *sagt:*  
29. NOVEMBER 2016 UM 18:48

Hi PoS,

As the other comment already mentioned your SAD ist still incomplete. Apart from that you were supposed to generate the first CRUD and have a demo of you first use case publicly available. Do you have something like that? I couldn't find a link or anything else pointing in that direction.

Kind regards  
SmartCinema

★ Gefällt mir

*Antwort*



OWNERTRAMP86.QOWAP.COM *sagt:*  
4. MAI 2017 UM 0:33

Simply on-line bank things out ... love the photos!

I attempt to learn by looking at other images, also.

<http://Ownertramp86.Qowap.com/3036428/fast-programs-for-gadget-insurance-compare-a-closer-look>

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 6. Dezember 2016 ~

## Gantt-Diagram

---

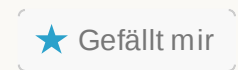
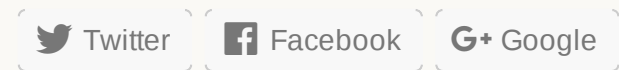
**T**HIS WEEK WE TRIED TO EXPORT OUR JIRA DATA TO CREATE a Gantt-Diagram but unfortunately it didn't work. So we created it by mostly by hand:

Gantt-Diagram [.mpp](#) / [.pdf](#) / [.gif](#)

We also had some changes in our [scope](#).

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

**DB-Schema & Class  
diagram**  
In "Allgemein"

**Testing 2**  
In "Allgemein"

**Semester Start**  
In "Allgemein"

PREVIOUS ARTICLE

*Software Architecture Document*

NEXT ARTICLE

*Semester Start*

## 3 GEDANKEN ZU “GANTT-DIAGRAM”



TINF15B4LSMF *sagt:*

8. DEZEMBER 2016 UM 0:11

Hello,

Your Gantt diagramm looks very nice, you even wrote in the assigned person, which wasn't really needed since according to the grading criteria „no handcrafting [was] necessary“.

Good work, keep it up,

## SilverScreen

★ Gefällt mir

*Antwort*



NEXTSMARTIDEA *sagt:*

8. DEZEMBER 2016 UM 9:52

Hey,

as requested you set up a MSProject. Your tasks have dates and you associated persons with them.

There are also milestones existing and you separated your tasks with phases.

Finally you did a good. But it would be better to have just one picture with all tasks.

Greetings,  
NextSmartIdea

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 5. April 2017~

## Semester Start

---

**F**IRST OF ALL YOU GET THE LINK TO OUR MIDTERM presentation: **Presentation**.

For the start of the new semester we thought of some little changes.

Technologie changes:

We will exchange our MVC Model from Spring to JSF because of easier



usage.

Scope changes:

## Use Case Diagram

Advertisements

Teilen mit:



Lade ...

## Ähnliche Beiträge

About us  
In "Allgemein"

Testing 2  
In "Allgemein"

Gantt-Diagram  
In "Allgemein"

## PREVIOUS ARTICLE

*Gantt-Diagram*

## NEXT ARTICLE

*Risk Plan*

## EIN GEDANKE ZU “SEMESTER START”

*Pingback: Final Presentation – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 11. April 2017~

## Risk Plan

---

**H** EY GUYS,

our first task is to find out what risks we have to avoid in our project:

### Risk Management

Next we tried to estimate our time spent on each use case.

## Use Case Estimation

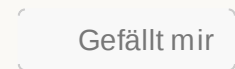
Our new use cases for this semester are:

- UC Cash Opening Entry
- UC Cash Check
- UC Close Cash Accounts
- UC Select between Management- and Cash- Area

Here you can find our whole **scope**.

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

**Final Presentation**  
In "Allgemein"

**Function Points**  
In "Allgemein"

**The Vision**  
In "Allgemein"

PREVIOUS ARTICLE

*Semester Start*

NEXT ARTICLE

*Function Points*

## 4 GEDANKEN ZU “RISK PLAN”



DANNYNATOR *sagt:*

11. APRIL 2017 UM 15:15

Hey guys,

great job you are doing there! A very convincing list of risks and nice ideas to mitigate them. The risk factor column looks somewhat odd to me though. The factor, which is calculated as (impact x probability) should normally end up somewhere between 0 and 10. If you measure your probability in percent the numbers must add up.

Good luck with your project!

Cheers,

Danny

★ Gefällt mir

*Antwort*



SAMS96LINA *sagt:*

19. APRIL 2017 UM 8:49

Hey guys,

you have made a great job! It's very good to have a good and detailed risk plan and your's looks very great! Your Use Cases are make with a lot of work that I can see.

Greetings

WhiteCode

★ Gefällt mir

*Antwort*



NEXTSMARTIDEA *sagt:*

19. APRIL 2017 UM 10:26

Hey guys,

your risk management looks good you have mentioned lots of points. According to the task it should be sorted by highest risk factor and you could give each risk a clear description.

According to the table with the time you spent on each use case you have implemented just two use cases?

Your use case documents are looking good and you have also some feature files which were not even necessary.

King regards,

NextSmartIdea



★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 24. April 2017 ~

## Function Points

---

W

E CALCULATED OUR FUNCTION POINTS FOR EVERY USE  
case with „Tiny-Tools„.

The following link shows our use case estimation added with the function  
points:

[Estimation](#)

This is our Complexity Adjustment Table:

## Complexity Adjustment Table

ITEM	COMPLEXITY ADJUSTMENT QUESTIONS	SCALE					
		No Influence 0	1	2	3	4	Essential 5
1	Does the system require reliable backup and recovery?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	Are data communications required?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	Are there distributed processing functions?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4	Is performance critical?	<input type="radio"/>	<input type="radio"/>	<input checked="" 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 type="radio"/>	<input checked="" 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 type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" 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 checked="" type="radio"/>	<input 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 type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
11	Is the code to be designed reusable?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
12	Are conversion and installation included in the design?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
13	Is the system designed for multiple installations in different organizations?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" 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"/>

**Here** you can see how we calculated everything:

2. For transactions (**EI**, **EO**, and **EQ**), the rating is based on the **FTR** and **DET**.

1. **FTR** - The number of files updated or referenced.
2. **DET** - The number of user-recognizable fields.
3. Based on the table below, an **EI** that references 2 files and 10 data elements would be ranked as **average**.

FTR's	DET's		
	1 - 5	6 - 15	> 15
0 - 1	Low	Low	Average
2 - 3	Low	Average	High
> 3	Average	High	High

3. For files (**ILF** and **ELF**), the rating is based on the **RET** and **DET**.

1. **RET** - The number of user-recognizable data elements in an **ILF** or **ELF**.
2. **DET** - The number of user-recognizable fields.
3. Based on the table below, an **ILF** that contains 10 data elements and 5 fields would be ranked as **high**.

RET's	DET's		
	1 - 5	6 - 15	> 15
1	Low	Low	Average
2 - 5	Low	Average	High
> 5	Average	High	High

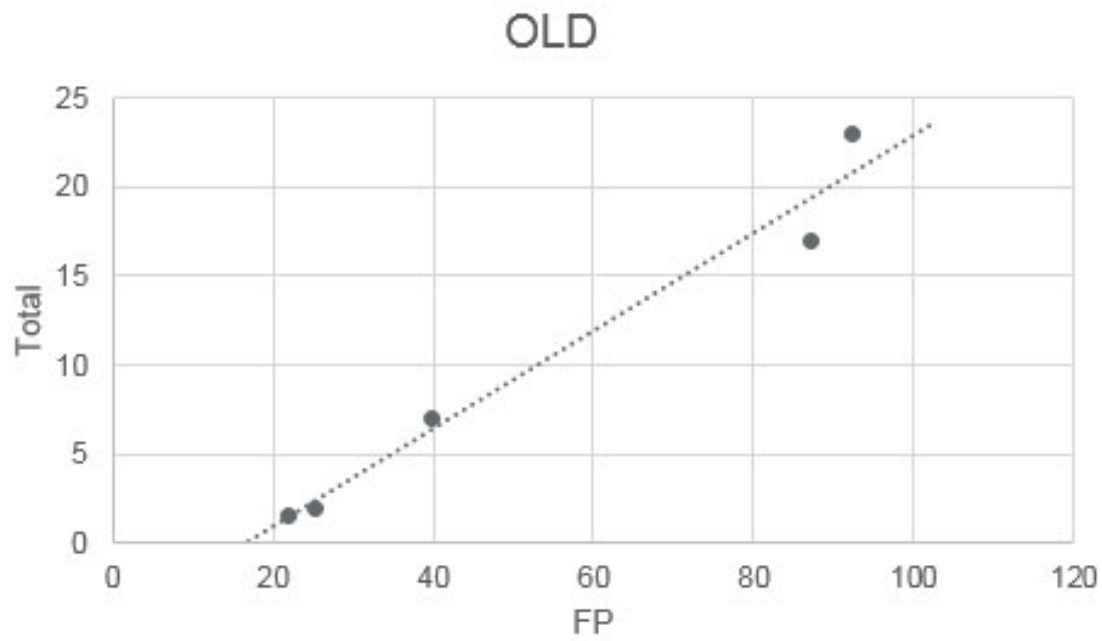
Here are our already implemented use cases:

- Selling Process
- Cash Check

There are the estimations we made for our following use cases

- Cash Opening Entry
- Close Check Accounts
- Select Between Management- And Cash-Area

As you see in our FP and Time calculation we have no explicit outliners:

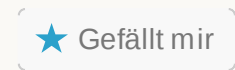
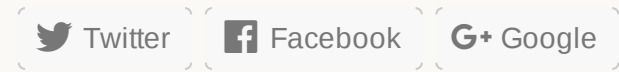


[Link to Excel Sheet](#)

Advertisements



Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

**Final Presentation**  
In "Allgemein"

**Risk Plan**  
In "Allgemein"

**Code Pattern**  
In "Allgemein"

PREVIOUS ARTICLE

*Risk Plan*

NEXT ARTICLE

*Testing*

## 5 GEDANKEN ZU “FUNCTION POINTS”



TINF15B4LSMF *sagt:*  
26. APRIL 2017 UM 9:10

Moin moin.

Looking good, looking good! What you should perhaps add, is an explanation about your weighting factors for the use-cases.

Other than that, if your estimation in the table up to are adjusted for the function points, the rest should be just fine.

Tschöööööö.

★ Gefällt mir

*Antwort*



TINF15B4LSMF *sagt:*  
26. APRIL 2017 UM 9:20

Just saw that you added the explanation about the weighting factors just now, now that's some quick reaction time. Well done, guess my critique is null and void now.

★ Gefällt mir

*Antwort*



SMARTCINEMAPROJECT *sagt:*

26. APRIL 2017 UM 9:16

Great work you did there! You showed all the calculations and the estimations for new use cases and I believe you completely fulfill the grading criteria.

★ Gefällt mir

*Antwort*



GAMESCOBOLD *sagt:*

26. APRIL 2017 UM 11:20

Hi, your blog entry looks very good and especially the graph is very nice. I agree with SMARTCINEMAPROJECT that you fulfill the grading criteria.

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – Written Handin – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 3. Mai 2017 ~

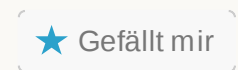
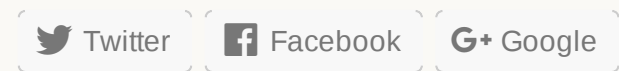
## Testing

---

**T** HIS WEEK WE'VE THREE NEW LINKS FOR YOU:

1. [Test code](#)
2. We don't have a screenshot, but we've already the [visualization of our tests](#)
3. [Maven file](#)
4. [RUP Test Plan](#)

Teilen mit:



Sei der Erste dem dies gefällt.

PREVIOUS ARTICLE

*Function Points*

NEXT ARTICLE

*Refactoring*

## 4 GEDANKEN ZU “TESTING”



GAMESCOBOLD *sagt:*

3. MAI 2017 UM 10:50

Hey PoS,

that's a rather minimalistic blog post, but you seem to have everything except a screenshot proving that you can run your tests in your IDE, you might want to add that still as it is part of the grading criteria. The Test Plan document also still needs some work, but that's understandable at this point, every group only has a draft version right now. Otherwise good job!

Cheers

★ Gefällt mir

*rAntwort*



POS SYSTEMS *sagt:*

3. MAI 2017 UM 11:00

Thanks for the comment. We haven't a screenshot, because we've already visualized our test, but unfortunately we forgot to mention it [] Now everything should be fine []

★ Gefällt mir

*Antwort*



SAMS96LINA *sagt:*

3. MAI 2017 UM 11:32

Hey PoS,

I can just justify the former comment!

Greetings,  
sams96lina

★ Gefällt mir

*Antwort*



## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 3. Mai 2017 ~

## Refactoring

---

H EY GUYS,

in our last lesson we learned some things about code smells and refactoring.

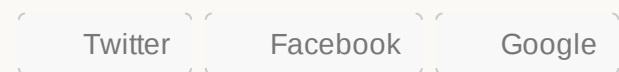
So we tried to learn and use it with the book **Refactoring**. For every step we made is a new commit that everyone can see what we did step by step.

Here you can find the links to our repositories:

<i>Name</i>	<i>Repository</i>
Sven Baumann	Fowler
Sandra Kramlich	SWE-Homework
Dominik Schneider	Fowler

Advertisements

Teilen mit:



Gefällt mir

Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

**Final Presentation**  
In "Allgemein"

**Metrics**  
In "Allgemein"

**Function Points**  
In "Allgemein"

◀ PREVIOUS ARTICLE

*Testing*

NEXT ARTICLE ▶

*Code Pattern*

## 4 GEDANKEN ZU “REFACTORING”



CHRIS *sagt:*

9. MAI 2017 UM 19:27

Hey there,

the refactoring you guys done looks pretty good. The commits you done are clearly seperated and don't touch to much code, so it's easy to follow your work step by step. The only thing im missing is, a little explanation why you guys done this refactoring projects. It would be very interesting. []

Kind regards  
smartCinema

★ Gefällt mir

*Antwort*



TINF15B4LSMF *sagt:*

10. MAI 2017 UM 10:15

Hey ,

According to the grading criterias your team has to provide one refactored project per member, which you have.

You have several commits, each with comment, while removing the code smells step by step.

This also seems to be in order and executed properly.  
Considering all this, your work is fully capable of receiving the full amount of points.  
Go get them!

Regards LSMF

★ Gefällt mir

*rAntwort*



MMIO *sagt:*

17. MAI 2017 UM 10:24

Hello,

everyone of you guys has linked a GitHub Repo and has enough commits. Your commit messages are looking good aswell.

Regards  
mMio

★ Gefällt mir

*rAntwort*

*Pingback: Final Presentation – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 13. Mai 2017 ~

## Code Pattern

---

**T**HIS WEEK WE LEARNED ABOUT CODE PATTERNS AND THEIR importance.

As homework we had to decide for a fitting code pattern for our project and to implement it.

Given our project, which is using the MVC model and realising a cash system, it wasn't that easy to find a fitting pattern. The best patterns would be



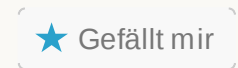
„Observer“ and „Singleton“, because of a static UI which uses a lot of dialog-boxes to interact with the user. But these two doesn't count in our SWE class so we searched for another one. Unexpectedly we found another one, the „Chain of Responsibility“, which was fitting quite well and was already implemented due to the „Count“ class. With this pattern, the classes should get much smaller.

Here you can see our class diagram.

Here you can see the code.

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

**Final Presentation**  
In "Allgemein"

**Metrics**  
In "Allgemein"

**Refactoring**  
In "Allgemein"

PREVIOUS ARTICLE

*Refactoring*

NEXT ARTICLE

*Metrics*

### 3 GEDANKEN ZU “CODE PATTERN”



SMARTCINEMAPROJECT *sagt:*

17. MAI 2017 UM 8:47

You chose a pattern that you had already implemented – which is fine, I think, although the grading criteria say that we are supposed to refactor existing code to conform to a new pattern.

Because you already implemented the pattern from the start, you logically can't have any before/after class diagrams. Which is unfortunate but I guess that's just how it is.

I'm missing a more detailed explanation of WHY the pattern fits so well here – are more smaller classes generally preferable over fewer bigger classes, or does the pattern reduce the overall code size?

It would also be very cool to have a link to the relevant code.

★ Gefällt mir

*Antwort*



JANIKRITZ *sagt:*

17. MAI 2017 UM 9:03

Everything smartcinema said is right and it would be great if the class diagram was a bit more readable, for example hide the getter and setters, so the classes are smaller and easier to understand.

Except for that you integrated the design-pattern good, and it looks like it will help you in the future.

Kind regards

Janik from FactRacing

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – Written Handin – PoS Systems*

*Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 24. Mai 2017 ~

## Metrics

---

**H**ELLO PEOPLE!

Today we present you our integration of a code metrics tool into our building process. A code metrics tool measures quality of code by a preset set of criteria and helps in writing more readable, more secure and easier to maintain code and warns you of very bad practises. Examples here were overriding the equals method of a java class whilst not also overriding the

hashCode method, which can lead to unexpected behaviour in hashmaps or other data structures, where the hashCode method is used to differentiate the classes.

The metrics tool of our choice is SonarQube as it is easy to integrate into maven builds and has many well defined code quality criteria. SonarQube works by analyzing code on a SonarQube server and showing the results on a website. For our project we used the dhw Karlsruhe SonarQube server to host the metrics of our code. Integrating it into maven is as easy as putting the SonarQube server's URL and the programming language our code is written in (i.e. java) as properties into our pom.xml and running *mvn sonar:sonar*. The results of our code analysis can be seen [here](#).

Of course we integrated our SonarQube workflow into our automated building process hosted by Travis-CI. Interested readers can view the state of our builds [here](#) and can even read building logs.

You can find our Test-Plan document [here](#).

Sonar showed us some issues with our code, so we had a better look at it.

Some things, which were criticized, were array lists with an explicit type. It said it would be more dynamic without it, but in our project it isn't necessary so we left it out.

Another point were the empty constructors in two classes:

```
public class Employee {  
    private int _id;  
    private String _name;  
  
    public Employee() {  
  
    }  
}
```

Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation. vor 23 Tagen L7 suspicious  
Major Open Not assigned Not planned 5min debt

```
public class ProductGroup {  
    private int _id;  
    private String _name;  
  
    public ProductGroup() {  
  
    }  
}
```

Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation. vor 23 Tagen L7 suspicious  
Major Open Not assigned Not planned 5min debt

The empty constructor is there, because it's meaningful to have it just in



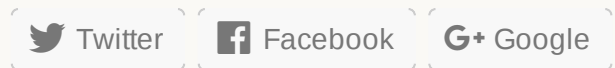
case. So, like suggested, we added a comment to explain why these constructors exist:

```
public ProductGroup() {  
    //just in case  
}
```

```
public Employee() {  
    //just in case  
}
```

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.  
Lade ...

## Ähnliche Beiträge

Final Presentation  
In "Allgemein"

Code Pattern  
In "Allgemein"

Refactoring  
In "Allgemein"

## PREVIOUS ARTICLE

*Code Pattern*

## NEXT ARTICLE

*Testing 2*

## 3 GEDANKEN ZU “METRICS”



GAMESCOBOLD *sagt:*



24. MAI 2017 UM 11:05

Hey,

your work this week is very good.

You almost pass all grading criterias. Only thing is that you missed the link to your testplan-document.

Also interesting is that the DHBW Karlsruhe SonarQube server is working for you, we had problems with it.

Kind regards

★ Gefällt mir

*Antwort*



TINF15B4LSMF *sagt:*

31. MAI 2017 UM 10:17

Hey guys,

your blog post seems to fullfill all grading criterias and everything was done correctly.

What is still missing is your testplan doc, which should be linked in this post but will be in the next post anyway as well.

Regards

LSMF

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – Written Handin – PoS Systems*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 31. Mai 2017~

## Testing 2

---

# H

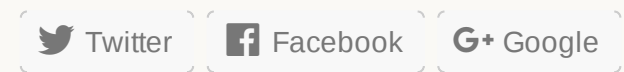
EY THERE!

We are using SonarQube to calculate our test coverage which is at 37% at the moment. It is recalculated on every push to our master branch.

Also, take a look at our **Test-Plan document** which describes the different tests we are using.

Advertisements

Teilen mit:



Lade ...

Ähnliche Beiträge

**Testing**  
In "Allgemein"

**Installation Test**  
In "Allgemein"

**About us**  
In "Allgemein"

PREVIOUS ARTICLE

*Metrics*

NEXT ARTICLE

*Installation Test*

## 3 GEDANKEN ZU “TESTING 2”



TINF15B4LSMF *sagt:*  
31. MAI 2017 UM 10:49

Hello guys,

A coverage of 37% is pretty godd but I think you linked the Testing plan of another group.

Greetings,  
Tinfl5B4LSMF

★ Gefällt mir

*Antwort*



MUMSFIT *sagt:*

31. MAI 2017 UM 11:16

Hello from MuM,

great job with the 37% code-coverage. However, a little proof would be nice ;).

We can also just find two tests in your Testplan, but we assume that you did not figure out which test to use for the third one. The same applies to us, so we are not in the right position to judge anyone.

Except for that, great job!

Kind regards, MuM.

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*



## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 14. Juni 2017~

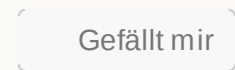
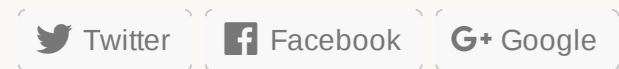
## Installation Test

W

**E HAVE PREPARED A DOCKER CONTAINER WITH**  
installation instructions in our Github Readme.

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

DB-Schema & Class  
diagram  
In "Allgemein"

Software Requirements  
Specification  
In "Allgemein"

First two Use-Cases  
In "Allgemein"

PREVIOUS ARTICLE

*Testing 2*

NEXT ARTICLE

*Final Presentation*

## 4 GEDANKEN ZU “INSTALLATION TEST”



SMARTCINEMAPROJECT *sagt:*

14. JUNI 2017 UM 11:32

We did it: <http://i.imgur.com/O6ttRBR.png>

★ Gefällt mir

*rAntwort*



SMARTCINEMAPROJECT *sagt:*

14. JUNI 2017 UM 11:33

Oh shit wrong link: <http://i.imgur.com/9StNoXj.png>

★ Gefällt mir

*rAntwort*



JANIKRITZ *sagt:*

14. JUNI 2017 UM 11:57

We did it too and it worked well.

Installation was fast, except for download time.

Prove: <https://imgur.com/a/99ba5>

You should note, that it is easy to lose money with negative prices.

Greetings FactRacing

★ Gefällt mir

*Antwort*

*Pingback: Final Presentation – PoS Systems*

*Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen

p

~ Veröffentlicht am 17. Juni 2017~

## Final Presentation

---

**H** EY GUYS,

our project is finished! Here you have an overview of our doing.

Have fun!

The PoS System Team! (Baumann Sven, Kramlich Sandra, Schneider

Dominik)

## Requirements:

UC

## Overall Use Case Diagram

And here are the links to each of our use cases:

- [Selling Process](#)
- [Cash Check](#)
- [Cash Opening Entry](#)
- [Close Check Accounts](#)
- [Select Between Management- And Cash-Area](#)

SRS

[SRS](#)

Test Cases

[Features Files](#)

Test Log

[Features Files](#)

Testcoverage

[Test Coverage](#)



Test Plan

Test Plan

Functional Test

Features Files

In Test Plan

Unit Test  
One other test

Unit Tests

User Interface Testing

**Project Management**

RUP gantt chart (past)

Gantt Chart

future long-term plan using FP estimation

Use Case Estimation

Graph for Estimation

Jira

Jira

Burndown

Burndown Example

FP calculation

FP Calculation

Function Points Blog

Ability to Execute

Demo

Do the **Installation** and try it yourself!

Code

Code in Github

Installation

Installation

Quality:

Architecture

DB Schema

SAD

Deployment Diagram

Pattern

SAD

Code Pattern Blog

Configuration Management/ Environmental Setup

Part in SRS

Metrics

Metrics Blog

Risk Management

Risk Management

Automated Testing

Travis for automated Testing

Automatic Deployment

Automated Deployment with Docker

Pattern

Code Pattern Blog

Other

Presentations

Midterm Presentation

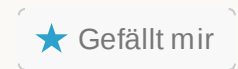
Blog Entries

- About us
- The Vision
- Technologies and Team Roles
- Software Requirements Specification
- First two Use-Cases

- Jira
- .feature
- DB-Schema & Class diagram
- Software Architecture Document
- Gantt-Diagram
- Semester Start
- Risk Plan
- Function Points
- Testing
- Refactoring
- Code Pattern
- Metrics
- Testing 2
- Installation Test

Advertisements

Teilen mit:



Sei der Erste dem dies gefällt.

### Ähnliche Beiträge

**Function Points**  
In "Allgemein"

**Risk Plan**  
In "Allgemein"

**Metrics**  
In "Allgemein"

PREVIOUS ARTICLE

*Installation Test*

## *Kommentar verfassen*

Gib hier deinen Kommentar ein ...

Erstellt mit WordPress.com.

Folgen