

Our vision

Mi, 04 Okt 2017 11:45:46, dhpoly, [category: allgemein]

Welcome to - DHPoly!

The only time that is wasted is the time when you think about time.
To not let that happen, we will provide you with the best entertainment there will be.

That is what our Project "DHPoly" is for.

Our Goal is to create a Monopoly everyone can enjoy.

First things first - it is going to be for free, meaning free access and no purchases.

Secondly - our Monopoly will be based on the usual Monopoly rules.

There are five major sectors we will be focussing on:

- Playing field: The playing field will be different from all the other Monopolys' playing fields, but it will contain all the mayor features one connects to a common Monopoly.
- Surface: A nice looking surface design invites to have a game or two.
- Events: Interesting and crazy events and dynamic game changes provide a completely new game experience.
- Trading and making Business: When playing Monopoly playing and making the smartest decision for your Business Play a very important role.
- AI (artificial intelligence): We want to make it possible, to Play Monopoly even when your friends are not around. Thatfor there will be the opportunity to Play against programmed robots.

That said, we are looking forward to every input you can give to us.

Because our goal is not just to program a good game, but to create a great Monopoly!

Best Regards

Your DHPoly-team :)

Semester 1: Week 2 - First determinations

So, 15 Okt 2017 23:59:13, dhpoly, [category: allgemein]

After the first week of focussing on our project we have specified roles in our team.

Role	Description
Implementer (Rico)	• implementations are not bound to the implementer - everyone on the team can write code
	• observe, that MVC (Modern View Control) patterns are used
	• make sure that the developments are leading towards our defined goals
	• keep an eye on all of the project's code
Requirements Specifier (Sven)	• executes Unit-Tests
	• decides about the importance of sub-processes
	• prioritizes tasks

Tool Specialist (Sven)	<ul style="list-style-type: none"> • is responsible for all the tools • decides about what tools are used
Tester (Sven)	<ul style="list-style-type: none"> • check if tests are reasonable and correct • check if tests are complete • makes sure the tests are executable and lead to a result
Designer (Alex)	<ul style="list-style-type: none"> • is responsible for the layout of the playing field and for changing it • it is his task to make the surface look unique
Blog poster (Alex)	<ul style="list-style-type: none"> • documents any progress in our project • reacts to comments and answers them • passes suggestions on to the project manager
Project manager (Alex)	<ul style="list-style-type: none"> • picks out the most useful proposals and shares these with the team • coordinates the time schedule of the team • creates a pleasant working atmosphere • distributes the given tasks

Technologies:

After clarifying our needs in terms of basic software features, we decided to use Swing MVC.

We want our source code to be easy maintainable, so that we can implement new features or fix bugs without investing a lot of effort - and as far as we are concerned, Swing MVC is the best choice.

For our version control we are using GitHub, because we have already gathered some experience with GitHub.

As our project management tool we are going to use TeamCity by JetBrains.

For further information keep following our blog.

(unpublished), dhpoly, []

[Software Requirements Specification](#)

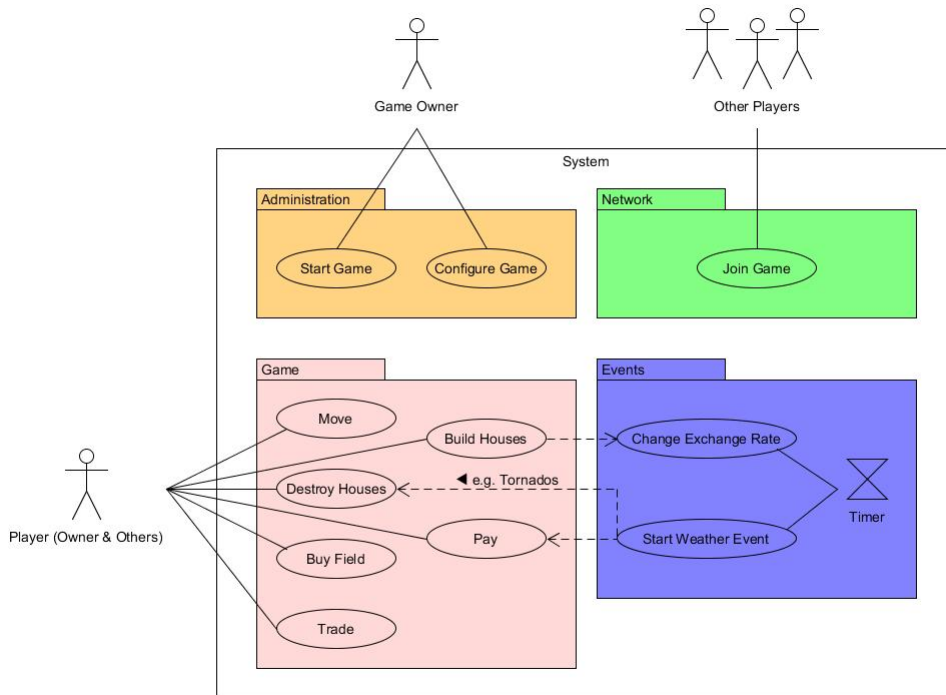
Semester 1: Week 3 - Software Requirements Specification

Mo, 23 Okt 2017 00:01:08, dhpoly, [category: allgemein]

In the last week the DHPoly-team focussed on describing the Software Requirements Specification and editing a document regarding those specifications. We will change the document many times throughout the development of the project. That's why it is not placed in a blog entry - please keep that in mind.

The described document can be found [here](#).

Additionally we built our Use Case Diagram which is shown below.



Semester 1: Week 4 - Use Cases and Prototype

Mo, 30 Okt 2017 00:58:48, dhpoly, [category: allgemein]

This week we worked on the specification of two of our Use Cases.

We focussed on "Trading" and "Withdrawing":

- [UC-Trading](#)
- [UC-Withdrawing](#)

Additionally we linked our Use-Cases in our [SRS](#) file.

Semester 1: Week 5 - Scrumming

So, 05 Nov 2017 21:15:15, dhpoly, [category: allgemein]

Scrum:

In the past week we created our own scrumboard, which is currently not filled with all the needed information. That will be done until the upcoming Wednesday.

ProjectManagementTool:

For continues integration we are currently setting up a TeamCity server in addition to our YouTrack scrumboard. We already connected our development IDE with YouTrack.

The link to our Scrumboard is the following one:

<https://dhpoly.myjetbrains.com/>

Sprints:

Our first [sprint](#) started today.

You can find the Burndown-Chart [here](#).

This sprint is going to last for one week.

Semester 1: Week 6 - Feature Files

Mo, 13 Nov 2017 14:31:50, dhpoly, [category: allgemein]

Last week we focussed on the feature files of two of our Use Cases.

These are as follows:

Trading:

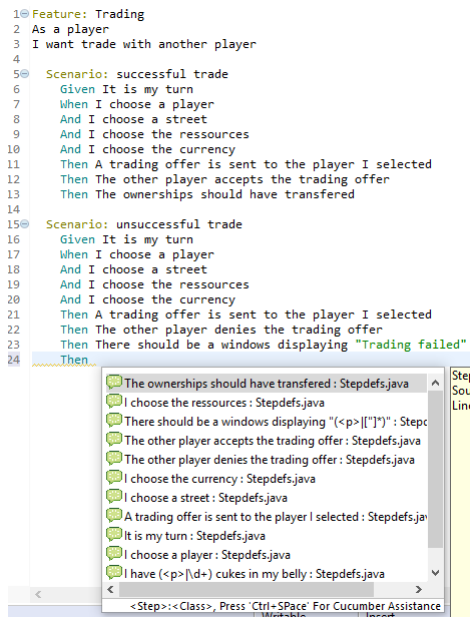
- [Use Case](#)
- [Feature-file](#)

Withdrawing:

- [Use Case](#)
- [Feature-file](#)

IDE features:

As you can see, that the auto completion, colour highlighting and formatting in our IDE are working properly.

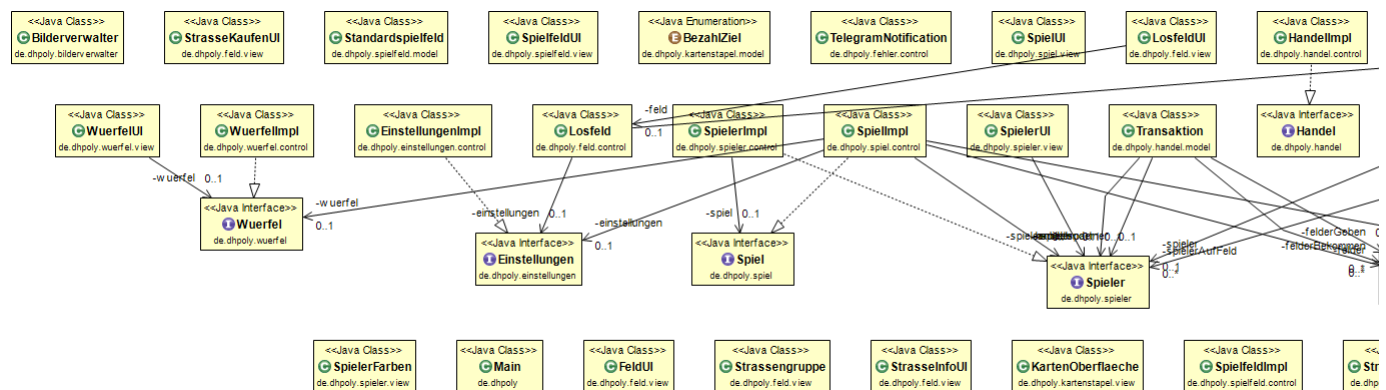


Semester 1: Week 7 - Class Diagram

Do, 16 Nov 2017 11:25:51, dhpoly, [category: allgemein]

In the last week we mainly worked on our class diagram. You can have a look at it [here](#).

The following picture is a version of the class diagram without methods and variables:



You can zoom into the preview, too.

The code the class diagram was generated of can be found [here](#).

To be able to create that diagram in our IDE Eclipse we used the UML plugin ObjectAid.

Information about that tool can be found on [this website](#).

Since we do not use a data base for our project, we can not show any data base scheme.

Semester 1: Week 8 - Architecture

Di, 28 Nov 2017 09:18:10, dhpoly, [category: allgemein]

This week we defined our software architecture.

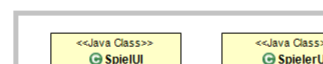
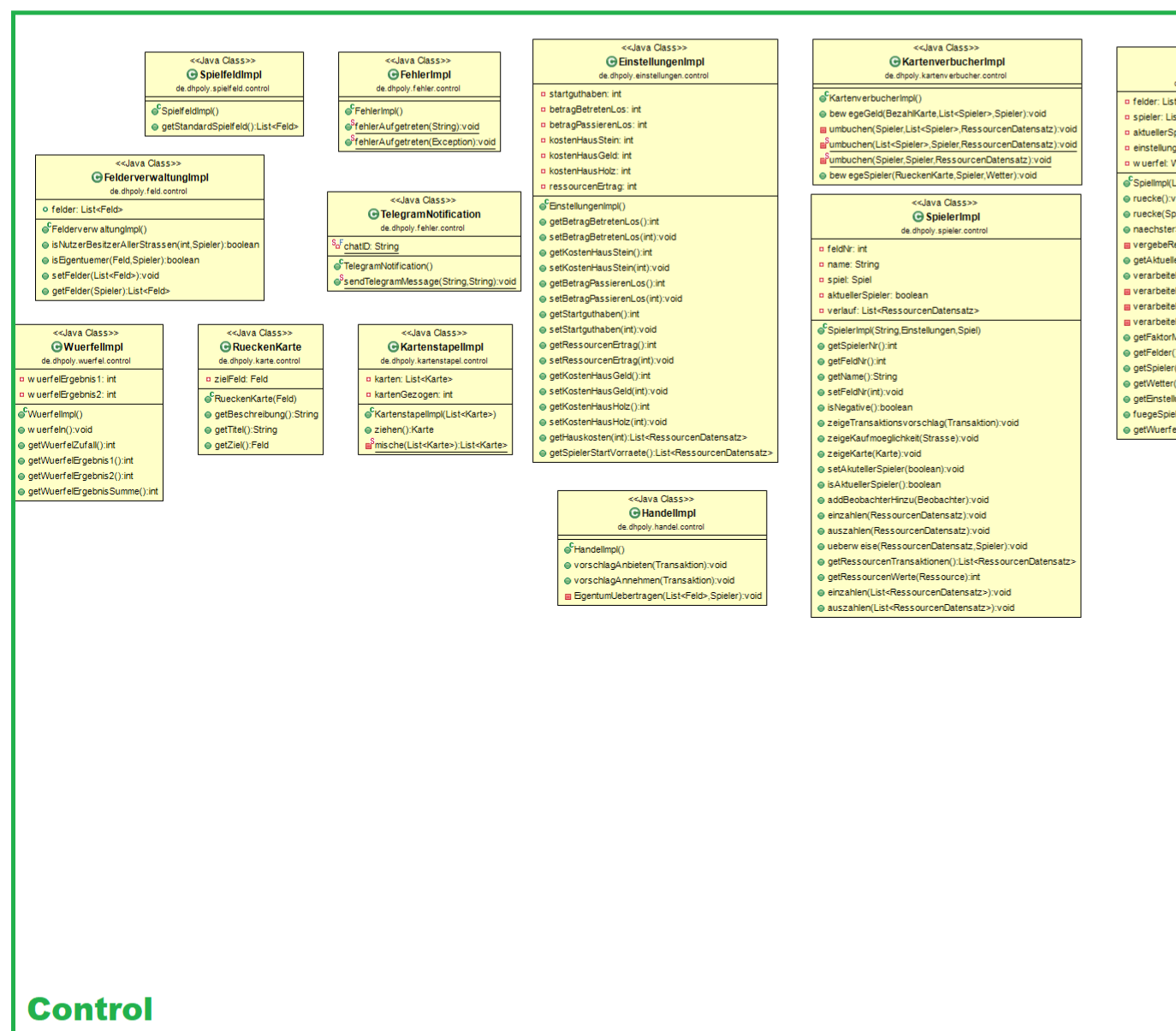
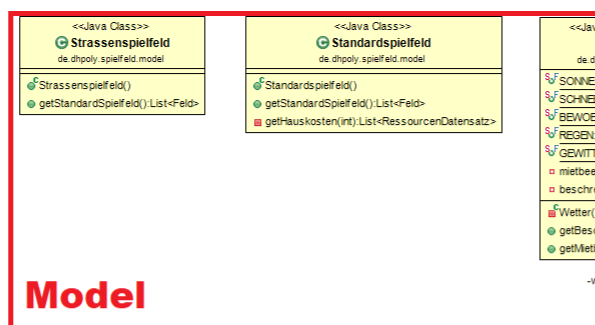
Demo version of our program / game:

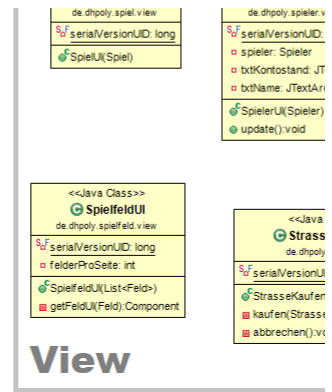
We have prepared a demo version of our current program.

To be able to execute this version, you have to follow [this link](#) to our GitHub. There you can download a zip file. you have to unzip that file, import it into Eclipse and start the java file.

Software architecture diagram:

The following is a preview of the diagram:





View

Logo:

Meanwhile we aslo created a Logo for our project and our group:



Semester 1: Week 9 - Gantt Chart

Mi, 29 Nov 2017 11:16:26, dhpoly, [category: allgemein]

In the last week we focussed on our Gantt Chart.

Since we are using YouTrack as our projectmanagementtool, we had a couple of problems.

First off the Gantt Chart created in Youtrack looks very confusing and does not give any information, because structure is lacking a lot.

Even for us, who know all the tasks in our project DHPoly, it is hard to read and understand this Gantt Chart.

Additionally You can not enter a finishing time when creating the chart. So YouTrack just adds up all the tasks in a row and sets the finishing date automatically (which creates a false end date). You can find our Gantt Chart [here](#).

But because this Gantt Chart does not give out the information we want it to, we also created a cumulative flow diagram.

This makes it possible, to overview the amount of work done and the amount of work that still has to be done (a feature that should be included in the Gantt Chart).

This diagram can be seen [here](#).

We would have liked a better looking Gantt Chart but we think the way we handled this problem, the outcomming information is acceptable and gives quite a good overview of our work.

If you have any questions or advises - please let us know. :)

Midterm Summary

(unpublished), dhpoly, [category: allgemein]

Semester 1: Week 10 - Midterm Summary

Mo, 11 Dez 2017 09:24:14, dhpoly, [category: allgemein]

After 10 weeks of hard work on our project we are going to take a break from DHPoly. The reason for that action is the switch of our workplace and therefore a change of our tasks and priorities.

But we do not want to leave just like that. For this reason below you can find a summary, that contains a link to every blog entry, which we created weekly.

Additionally we provide you with all the interesting information and data that lies within our blogs below.

Blogentries:

- Week 1: [What is DHPoly? + Our Vision](#)
- Week 2: [Role Assignment in our Team](#)
- Week 3: [Software Requirements Specification](#)
- Week 4: [Use Cases](#)
- Week 5: [Project Management Tool + Scrumming](#)
- Week 6: [Feature Files](#)
- Week 7: [Class Diagram](#)
- Week 8: [Software Architecture](#)
- Week 9: [Gantt Chart](#)

Documents:

- [Software Requirements Specification Document](#)
- [Software Architecture Document](#)

Important Links:

- [Our Github](#)
- [Our Code](#)
- [Our Logo](#)
- [Our Use Case Diagram](#)
- [Our Class Diagram](#)
- [Our Youtrack](#) (Projectmanagementtool)
- [Our Gantt Chart](#)
- [Our Cumulative Flow Diagram](#)
- [Team Member Time Report](#)
- [JUnitTest](#)

Burndown Charts:

- [Week 1](#)
- [Week 2](#)
- [Week 3](#)
- [Week 4](#)
- [Week 5](#)

Demoversion:

A demoversion of our project is downloadable [here](#). For further information we refer to our [blog entry](#) from Week 8.

MidTerm:

Click [here](#) to get to our MidTerm Presentation.

For further questions please leave a comment. :)

Your DHPoly-Team :)

Semester 2: Week 1 - Long Break, Big Expectations!

Mo, 09 Apr 2018 11:45:04, dhpoly, [category: allgemein]

Welcome back to our weekly blog,

after a 'short' break of about 12 weeks. :)

Now that we got back to work, we had to regain all the knowledge and information we once had.

That contains:

- Members: Have the project's members changed?
- Technologies: Which technologies are/will we be using?
- Scope: What is our current scope?
- Status: What is the actual Project Status?

Members:

The Project members have remained unchanged and so did their roles.

The members are as follows:

- Alexander Busch
- Sven Köhler
- Rico Kritzer

Technologies:

The technologies we use have not changed, too.

Scope:

Our scope is mainly equal to what has been before. The only change we had to make is, that we added the feature 'Multiplayer'. We want multiple people to be able to join one game. That will be possible on the same device or via LAN on different computers.

Status:

The Status is stated easily. The following Features are already running and usable:

- Dicing
- Moving
- Buying fields
- Buying houses
- Upgrading houses
- Trading
- Gathering ressources (such as wood and Stone)

Best regards

DHPoly

Semester 2: Week 2 - Risks

So, 15 Apr 2018 23:24:09, dhpoly, [category: allgemein]

Hi everyone :)

In the last week, which was the first week after our break, we had our first sprint issue.

Due to organisation problems and duties with higher prioritization we decided to do a two-week-sprint.

Besides that, we created a table that displays the time, we used for the creation of each of our use cases.

This can be seen [here](#).

buy field		8h45m	8h10m
DHP-117	Buy field	1h	1h
DHP-37	Interaction with unsold streets	1h	1h
DHP-38	Method: is Street sold	15m	15m
DHP-64	StraßeInfoUI design	3h	1h55m
DHP-60	UI for buying streets	1h	1h45m
DHP-25	UI for streets ownership	2h30m	2h15m
buy house		5h15m	4h45m
DHP-67	Build Houses	15m	20m
DHP-68	Build/Destroy houses UI	2h	1h35m
DHP-116	Buy house	1h	1h
DHP-68	Destroy houses	15m	10m
DHP-7	Implement Ressource UI	1h30m	1h30m
DHP-69	Sell Houses	15m	10m
moving		3h	3h10m
DHP-118	Moving	1h	1h
DHP-2	Moving on the field	2h	2h10m
trading		10h40m	11h20m
DHP-111	Get the Trading feature files running	2h	2h30m
DHP-51	Migrate .docx Trading UC to .md file	20m	20m
DHP-87	Offering Ressources UI	2h30m	2h20m
DHP-86	Offering Streets UI	2h30m	2h10m
DHP-127	Trading Button	20m	30m
DHP-16	Trading UI	3h	3h30m
withdrawing		5h20m	5h
DHP-8	Add bank account UI	1h	1h
DHP-112	Get the Withdrawing feature file running	2h	2h30m
DHP-52	Migrate .docx Withdrawing UC to .md file	20m	20m
DHP-84	Reflector BezahlKarte	2h	1h10m

Furthermore we have thought about the risks concerning our project and listed these.

You can see these here:

Risk Name	Risk Description	Risk Probability of Occurrence	Risk Impact	Risk Factor	Risk Mitigation	Person in Charge of Tracking
Members prevented from work	Due to private problems or more important duties, members can not invest the needed amount of time	30%	7	2,1	We try to plan our sprints and the contained tasks beforehand, so we can easily	Alex
Timing	We didn't plan our shedule correctly. Therefore the product isn't finished on time.	15%	8	1,2	Check regulary if the shedule is still correct and plan with time buffers.	Alex
AI can not be finished	The artificial intelligence has a specific purpose in our project.	20%	4	0,8	Luckily, if the developement of the AI fails, the game is still	Rico
Server can not be configured and set up in time	If setting up a fitting server and configuring it is either too dificult or takes too much time, it might appear, that we have to handle this task later on. This would mean, that	15%	3	0,5	To prevent that we dipatch a decend time budget for this purpose.	Alex
Quality	The quality of our game is insufficient.	5%	6	0,3	Beta test the game with multiple players.	Sven
Licensing	License problems with pictures we used or because of our game in general.	3%	10	0,3	Read licenses and check if we are allowed to use all images we implemented.	Sven

For the original file please click right [here](#).

Additionally we are planning on adjusting our use case time table. Stay tuned for that.Best regards,

DHPoly

Semester 2: Week 3 - Function Points

Mo, 23 Apr 2018 01:38:28, dhpoly, [category: allgemein]

Hi everyone :)

After several weeks of working on this project, we have learned more and more how about project management is. Time plays a huge part here, specifically speaking time management.

To improve our time management, it is very helpful to know, how long which task will take. Of course you can not specify the estimated time for everything, but, especially for use cases, we are capable to do so.

In fact, we are now using function points.

Function points are values, which are created from use cases and belong to these.

There is one function point (value) for one use case.

The values vary based on the complexity on a use case. In fact, the more complex a use case is, the higher its function point will be.

The way we can estimate time for future use cases is the following:

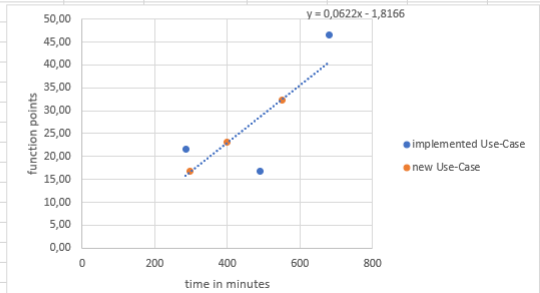
- create a few use cases (it was three for us)
- evaluate their function points
- create a diagram
- in there, one axis is the time and one is for the function point values
- now you put your use cases in there, which the function points and the time, it took to implement them
- connect all your diagram points
- now you have a graph, that displays, how long a use case with function point value X will take to implement
- create a new use case and evaluate its function point value
- have a look at your diagram, and you can see the estimated time

To evaluate function points we used [this](#) tool called tiny tools. In there we had to specify a number of statistics concerning the use cases and specificate, whether user inputs and outputs and more conditions are simple or complex. After filling out the needed data, all we had to do is to calculate the function points.

Have a look at our diagram:

Use Case	Estimated time	Spent time	Function Points	
buy_field	525	490	16,74	implemented Use-Cases
buy_house	315	285	21,63	new Use-Cases
trading	640	680	46,64	
event_field	298		16,72	
trading_ressources	550		32,40	
upgrade_house	400		23,04	

$x = 16,077 * y + 29,206$
 $x = \text{time}; y = \text{fp}$



The diagram can be also seen in our repository [here](#), such as the original excel file (we created the diagram in excel) [here](#).

Best regards,

DHPoly-Team

Semester 2: Week 4 - Testing

Do, 03 Mai 2018 16:16:46, dhpoly, [category: allgemein]

Hello guys :)

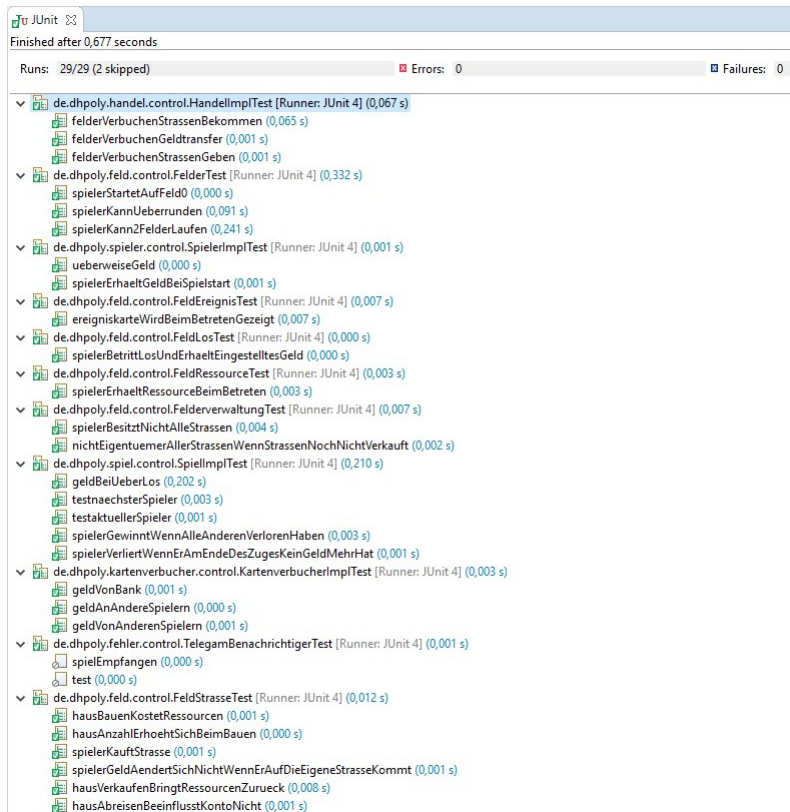
Welcome back to our weekly update.

This week we did a lot of research regarding testing.

You can find a link to our test code in our repository [here](#).

Because we are not using Maven at all, or anything like it, we do not have any maven files to show - unfortunately.

The testing in our project works fine at the moment, which can be seen in the following picture of a test:



To get more information about our testing, our motivation for testing and our future plans for testing, click [here](#) to see our testplan document in our repository.

Additionally we specified two new use cases.

These are:

- [Upgrade house](#)

- [Event field](#)

Click on the use cases to get to our repository.

Best regards,

DHPoly-Team

Semester 2: Week 5 - Refactoring

Mo, 14 Mai 2018 04:06:11, dhpoly, [category: allgemein]

Hi guys :)

This week we put our focus on refactoring code.

You can see examples of that from each of our members right here:

- [Alex](#)
- [Rico](#)
- [Sven](#)

Additionally we made use of Codacy. Here are our links regarding that:

- [Alex](#)
- [Sven](#)
- [Rico](#)

Refactoring code is very important to keep the code clean and readable, which leads to editable code.

But besides changing the code to bring it to a better state, it is also extremely valuable to test your code after nearly every change. That way you can be sure to not destroy your code, instead of improving it.

The three of us used the IDE Eclipse to refactor the code.

The probably best or and most used feature for Eclipse had to offer for us is the renaming, which can be simple done by pressing alt + shift + r.

Best regards,

DHPoly-Team

Semester 2: Week 6 - Patterns

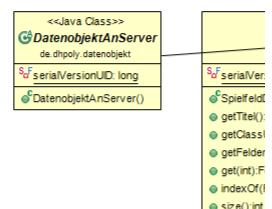
Mo, 28 Mai 2018 02:40:13, dhpoly, [category: allgemein]

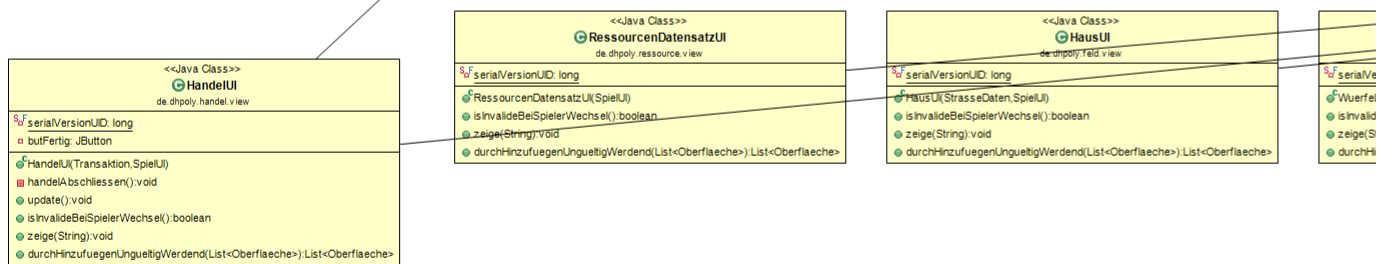
Hi,

the recent days were all about patterns.

In our project we use the "Factory-Pattern".

You can see a class diagram which our highlighted pattern [here](#):





The reason for our pattern decision is the following:

By using the factory pattern we can use factory methods. These methods offer one handy feature: With these we are able to create objects, without having to declare the exact class, the object is referring to.

Additionally, we will add a link to our repository where you can (kinda) see the implementation and use of our pattern in one class:

[Sample class](#)

Stay tuned! :)

Semester 2: Week 7 - Metrics

Mo, 04 Jun 2018 12:59:09, dhpoly, [category: allgemein]

Hi guys,

with a slight delay in time today we are talking About metrics.

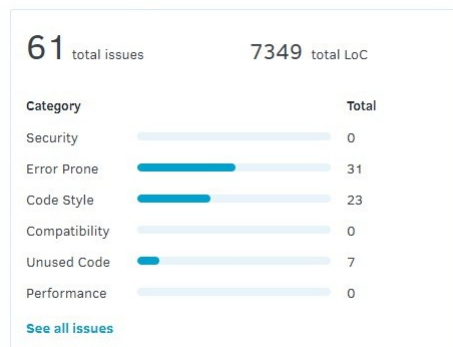
Now metrics are a very good way to monitor your projet's code and analyze its statistics.

We use the tool Codacy to create metrics for our code.

Furthermore we utilize SonarLint to improve our code's quality.

The following is a picture from Codacy, showing how many errors and flaws our code contains.

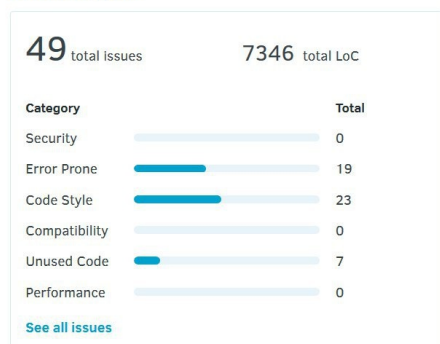
Issues breakdown



The most important "feature" here for our project is performance, because we are programming a game. And since there are no issues regarding the performance, we are pretty happy with this one. :)

After some work we were able to update our issues:

Issues breakdown



Now as you can see here, the total amount of issues has been reduced from 61 to 49.

The remaining ones are Error Prone, Cody Style and Unused Code. The problems regarding Code Style and Unused Code are caused by our Cucumber files / tests.

The Error Prones are the ones we are working on currently.

Down below you can see our codeclimate complexity rating:

Breakdown

247 FILES

MAINTAINABILITY

TEST COVERAGE

Codebase summary

MAINTAINABILITY

A 2 days

TEST COVERAGE



Repository stats

CODE SMELLS

6

DUPLICATION

8

After several requests we will Show a bit of Code representing how using Codacy our code's Quality improves:

```
4 src/test/de/dhpoly/feld/control/StrasseKaufenLogikTest.java View v
@@ -15,6 +15,7 @@
15 15
16 16 public class StrasseKaufenLogikTest
17 17 {
18 18 + private Datenobjekt gesendetesObjekt;
19 19 private StrasseKaufenLogik logik = new StrasseKaufenLogik();
20 20 private Spieler spieler = new Spieler("Peter");
21 21 private Spieler aktuellerSpieler = new Spieler("ba");
@@ -53,7 +54,6 @@ public void strassenKaufSendetStrasseAnClient()
53 54 assertTrue(gesendetesObjekt instanceof SpielfeldDaten);
54 55 }
55 56
56 56 - private Datenobjekt gesendetesObjekt;
57 57 private Spiel spiel = new SpielUnimplemented()
58 58 {
59 59 @Override
@@ -77,7 +77,7 @@ public SpielfeldDaten getSpielfeld()
77 77 @Override
78 78 public boolean kannSpielerStrasseKaufen(Spieler spieler, StrasseDaten strasse)
79 79 {
80 80 - return aktuellerSpieler == spieler;
81 81 + return aktuellerSpieler.equals(spieler);
82 82 }
83 83 };
```

In this example we used the "Equals" method instead of a simple equals operation (==). Furthermore we had global variables defined somewhere in the class, which are now at the start of the class.

This week's blog is a bit short caused by other tasks taking our work time. We apologize and hope you will be back for our next blog entry, which will be much more informative and interesting. :)

Best regards,

DHPoly-Team

Semester 2: Week 8 - Installation

Mo, 11 Jun 2018 11:19:30, dhpoly, [category: allgemein]

Hi guys,

This week it has finally come to the Point, where we have to install our project's application on a strange Hardware.

So we would like you, our Readers, to kindly download our zip archive and run the executable *.jar file.

You can download it here: [Download](#)

That should start the DHPoly application.

We would be happy to hear your results, experiences and advices.

Best regards,

DHPoly-Team

Semester 2: Week 12 - Final

So, 01 Jul 2018 01:47:32, dhpoly, [category: allgemein]

Hi guys,

today we proudly present you our final Blog entry.

Use Cases

- [withdrawing](#)
- [buy_field](#)
- [buy_house](#)
- [event_field](#)
- [trading_ressources](#)
- [trading](#)

- [upgrade house](#)

cucumber tests/Test Cases

- [Stepdefs](#)
- [trading](#)
- [withdrawing](#)

Tests general

- [JUnit Tests Screenshot](#)
- [Test Plan](#)
- [TeamCity \(screenshot of working TeamCity\)](#)

Software Requirements Specification

- [SRS](#)

Project Management

- [YouTrack](#)
- Burndown Charts:
 1. [Construction 1](#)
 2. [Elaboration 1](#)
 3. [Elaboration 2](#)
- [Time Report](#)
- [Gantt Chart](#)
- Functions Points ([screenshot/excel](#))

Demo/Installation/Code

- [Installation: Semester 2: Week 8 – Installation](#)
- [Code \(GitHub\)](#)

Software Architecture Document

- [SAD](#)

Metrics

- [Semester 2: Week 7 – Metrics](#)

Risk Management

- Risks ([screenshot/excel](#))

Pattern

- [Semester 2: Week 6 – Patterns](#)

Presentation

- [Final Presentation](#)
- [Handout](#)

Blog posts: Semester 1

- [Semester 1: Week 2 – First determinations](#)

- [Semester 1: Week 3 – Software Requirements Specification](#)
- [Semester 1: Week 4 – Use Cases and Prototype](#)
- [Semester 1: Week 5 – Scrumming](#)
- [Semester 1: Week 6 – Feature Files](#)
- [Semester 1: Week 7 – Class Diagram](#)
- [Semester 1: Week 8 – Architecture](#)
- [Semester 1: Week 9 – Gantt Chart](#)
- [Semester 1: Week 10 – Midterm Summary](#)

Blog posts: Semester 2

- [Semester 2: Week 1 – Long Break, Big Expectations!](#)
- [Semester 2: Week 2 – Risks](#)
- [Semester 2: Week 3 – Function Points](#)
- [Semester 2: Week 4 – Testing](#)
- [Semester 2: Week 5 – Refactoring](#)
- [Semester 2: Week 6 – Patterns](#)
- [Semester 2: Week 7 – Metrics](#)
- [Semester 2: Week 8 – Installation](#)

Blog as PDF

- [PDF download](#)

Best regards,

Your DHPoly-Team