BlogBooker

onetouch940978896.wordpress.com

Contents

1.   2019

1.1   October

Week 1: Getting Started (2019-10-02 10:47)

Hey All!

We would like to introduce you to our Vision. A one touch Cocktail machine.... but with a social touch. We would not only like to offer the Hardware, that means the "Sauf-O-Mat", but also an Application which allows you to customize your drink and see activities of your friends. Share your favorite drinks and get together!

Our team consists of 3 members: Anna Janzen, Tim Koch and Elia Herzog. We all share the same passion, which is getting drunk af. No just kidding of course we don't. But there is a little truth in that. We acknowledged that especially in IT people are used to not talk with each other... and a nice evening with drinks could sometimes maybe change that.

We are presenting you....

The "Sauf-O-Mat"

-

Stay tuned to get some more information soon.

The "Sauf-O-Mat" Team

Arerrac (2019-10-09 11:15:45)

Hello guys, i think its a great project to be accomplished. I personally know about the social problem of computer scientists. So with the help of your system you may be able to hop over your own barrier and get in touch with new people. Did you already break your architecture into realizable pieces? I think it will make it easier for you to reach your goal. How do you plan to get the fluid in the right amount into the machine? Greetings, CAD

annoua (2019-10-11 14:13:42)

Hey CAD, thanks for your Feedback! We just made our issue list and thought about smaller pieces of our project. The link to our issue list ist on the next blogpost. We will use pumps and control them via USB to get the fluid in the right amount into

the machine? Best Regards, the Sauf-O-Mat Team

logicgametinf18b3 (2019-10-09 11:42:04)
Hey, I enjoyed reading your blog and I also like that sketch of your "Sauf-O-Mat", it gives a small impression of how it will look in the future. I would like to know if you are going to seperate your blog entries. Are you going to leave every entrie on the front page or are you going to seperate every entrie on a single site? In my opinion you have not introduced yourself detailed enough. I think it would be great for other readers to know where you work, in what field you study or what your speciall skill in this project is. Best wishes, Marcel

annoua (2019-10-11 14:15:36)
Hello Marcel, thanks for your feedback. We just reorganized our blog, does it look better for you? Best regards, The Sauf-O-Mat team

Week 2: Roles and Tools (2019-10-09 12:06)

We distributed the following Roles based on the RUP standard to our team members as you can see. Maybe might have to change this a little in the future.

- Project Manager: Elia

- Tool Specialist: Elia

- Deployment Manager: Tim

- Designer: Anna

- Configuration Manager: Tim

- Implementation:

  - Frontend-Developer: all of us
  - Backend-Developer: all of us

- Test Manager: Anna

- Tester: all of us

- Blogposts: Anna

- Blog-Reviews: all of us

We want to realize the cocktail machine with a rasberry pi and create a case out of wood (maybe with a little help from mechanical engineers in the house), because we think it's the easiest way. We will use Python as programming

language and Visual Studio Code as IDE.

We are planning to use C # as our main programming language. Therefore we all use Visual Studio as IDE. Xamarin for Visual Studio allows us to implement the code for Android and iOS apps with a single code base.

We use github to share our code. Currently we also use it as our project management tool but we try to figure out, if it would be better to use You Track. At the moment we don't know how to use it correctly. Maybe someone of you could give us a little help?
Here you can see our issue list on [1]YouTrack.

1. https://onetouchnextgen.myjetbrains.com/youtrack/dashboard?id=5a3ea5e0-9f57-44b5-8273-cebf57afc28e

---

partyplayerteam (2019-10-09 16:36:23)
Hey Team "Sauf-O-Mat", It's really cool that your project consists not only of software, but also of the machine with the raspberry pi itself. I'm very excited how you will realize the hardware part. I also really like the idea of using xamarin for having an ios and android app with the same code base. But you haven't mentioned until now which tool for project management you will use, maybe you can add that to your blog post. Then you can also add a link to your issue list and explain how you will classify the issues regarding phase and workflow with your tool. You wrote that only one of you guys will be the main developer for front and backend. Maybe you can divide the work a bit more among yourselves. If you just wanted to express that he will do a big part of the implementation, you should add the other group members to the point "implementation". Best regards, partyplayerteam

annoua (2019-10-11 14:08:28)
Hey partyplayerteam, thanks for your feedback. We changed a few things you mentioned and we will also add a link to our issue list. It's still in the making. Best regards, the Sauf-O-Mat team

Project: Itemize (2019-10-13 11:52:23)
Hi guys, It's interesting and nice to see, that you have a nearly strict role differentiation. Hope you don't have troubles to obey on these. But I have to say. that your article seems a little incomplete, if you have a look at the GC. "- Link to your current issues list - make sure its public" "- Each issue should be classified by: Phase and Workflow. (This is different for each tool, so explain how it is done in your tool in your blog entry.)" "- RUP terminology used (link should help with this task)" You may have the roles of RUP, but you never mentioned that you orientated yourself by RUP. Also I miss why you decided to use your technologies or which criterias you had in mind, when you made your other decisions, like your wodden case. For yourself it might seem clear, but for the reader there's missing information. This should not only critizize your blog entry, but also help you to improve it and therefore the experience for the reader :). Best regards Millo@project:Itemize

annoua (2019-10-30 11:12:58)
Hey Millo, thanks for the feedback. We changed some thinks you mentioned and now our post should be complete. Regards, Anna

Midterm: Written Handin – One Touch next Generation (2019-12-21 11:53:51)
[...] BlogpostApplication on Pi:-Visual Studio Code-PythonApplication on mobiles:-Visual Studio-Xamarin-C #database:-Windows Server-MSSQL database [...]

Final – One Touch next Generation (2020-06-30 10:22:50)
[...] Blogpost Week 2SRS [...]

Week 3: SRS & UCD (2019-10-14 15:26)

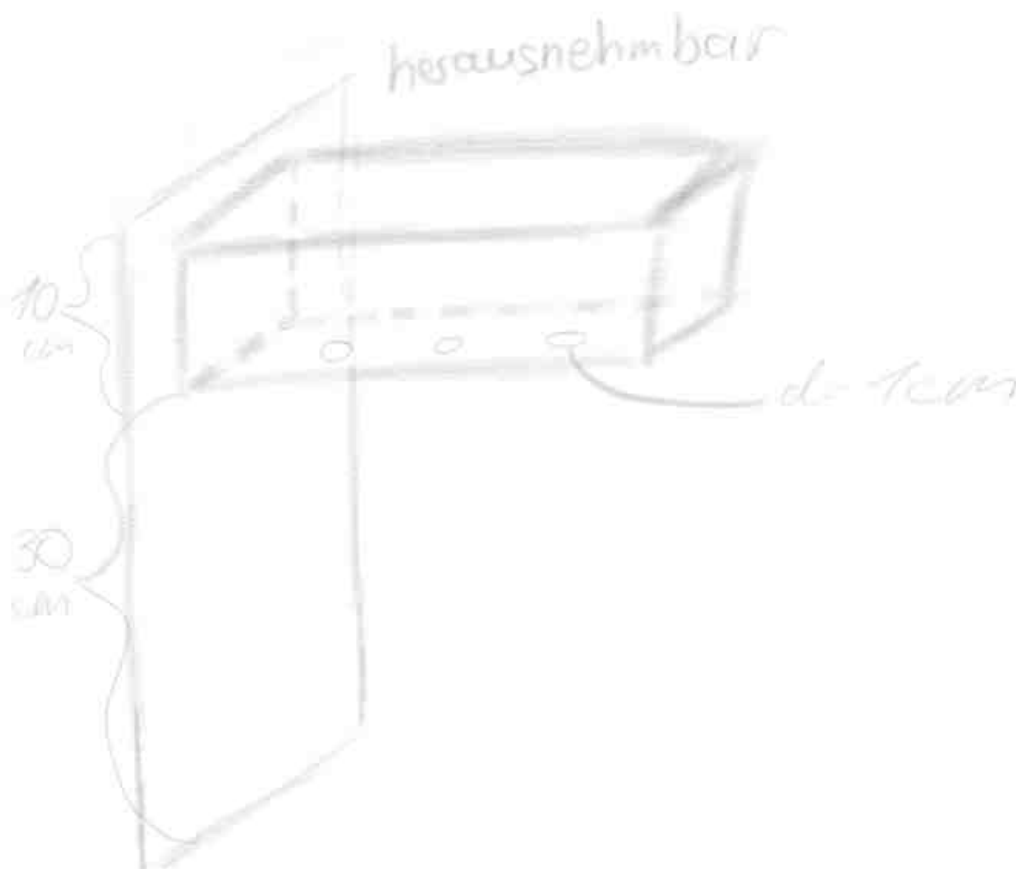In this link you can see our use case diagram:
[1]https://github.com/TheLordXII/OneTouch/blob/master/UseCaseDiagram.png

And in this link you can see our Software Requirements Specification. We will add more in the future.
[2]https://github.com/TheLordXII/OneTouch/blob/master/SRS.md

In the gallery you can see how the case will look like.

-

-

-

1. `https://github.com/TheLordXII/OneTouch/blob/master/UseCaseDiagram.png`
2. `https://github.com/TheLordXII/OneTouch/blob/master/SRS.md`
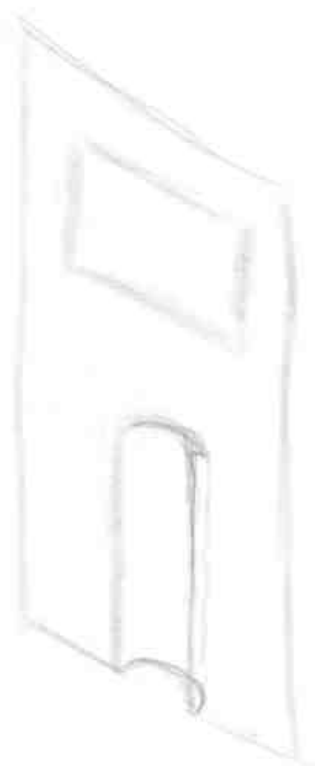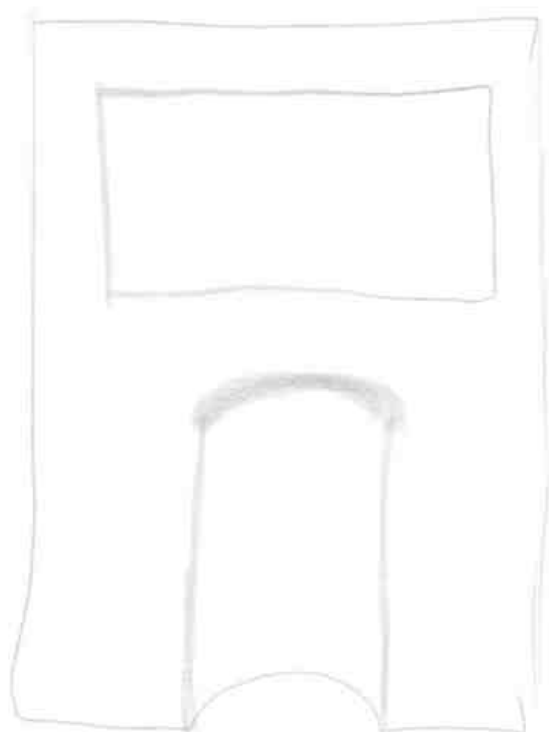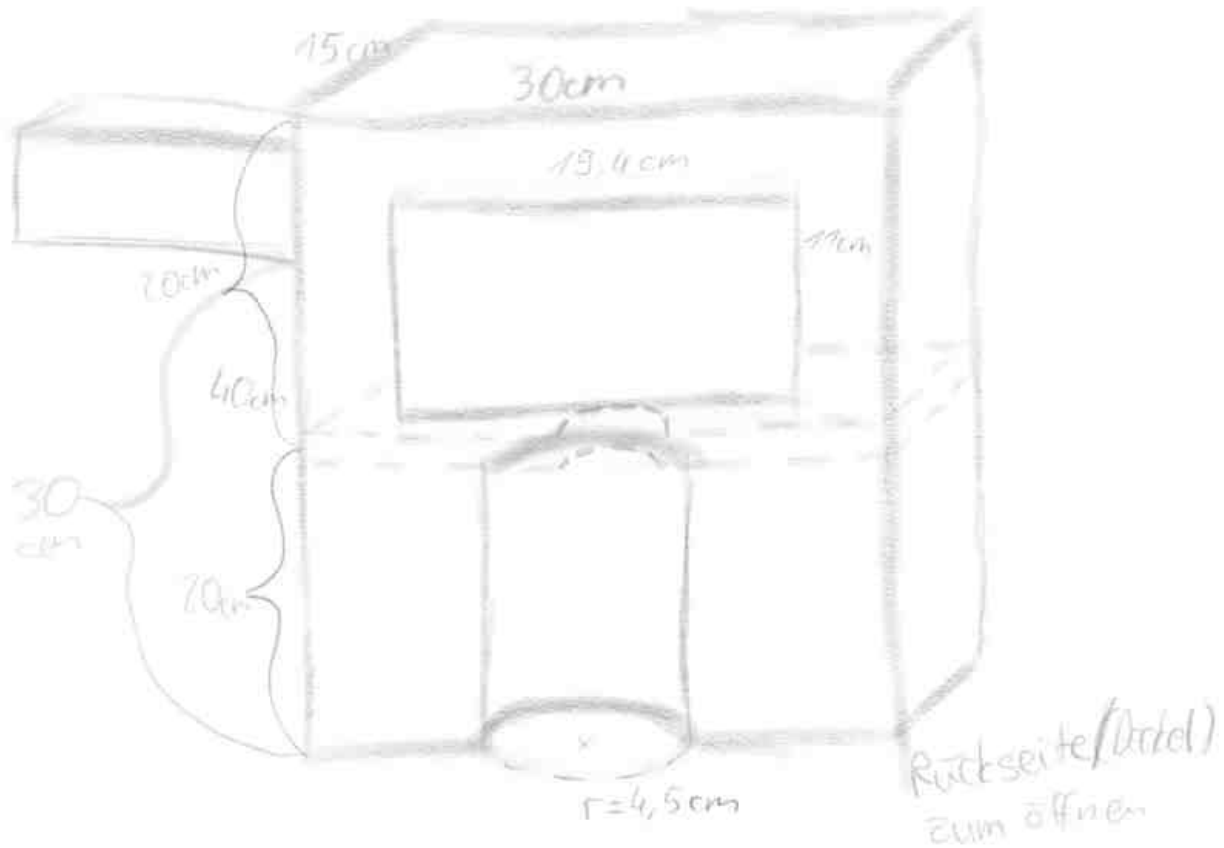
The hopper team (2019-10-20 17:18:57)

Dear OneTouch-Team, we really like your idea. To have cocktails at any time with just a press of a button is just awesome. First of all, in regards to your UCD: We think, you could have prettified the diagram a bit, it looks a bit messy. Also, why is "logging in" a use case for a "logged in user"? And why is an administrator using a user? Also, you have no dependencies and legend for the colors is missing. Secondly your SRS seems rather incomplete. Only one paragraph is filled. Could you replace the blue texts, either with content or with "tbd."? Best regards,

annoua (2019-10-30 11:16:28)

Dear the hopper team, we redesigned our UCD a bit so now it looks nicer. We also filled out some more paragraphs in our SRS. Regards, Anna

Fridgify Team (2019-10-20 19:38:32)

Dear OneTouch-Team, we, the fridgify team, are just as in love with this idea as the hopper team and would like to be your first customers. If you are planning to offer the local use of the "Sauf-O-Mat" and the use through the app, we would redo the UCD and split it up to two services eg. first Sauf-O-Mat-Machine, second Sauf-O-Mat-App and create more specific functions and connections. Secondly, we would like to re-read the SRS as soon as it is done. So we can't say much to it right now. As far as we can see there are still plenty of empty fields and certain parts from the template, which should be removed. We are eagerly looking forward to hearing more from you and an answer as soon as the document is finished. Stay cool. Stay organized. — Your Fridgify Team

annoua (2019-10-30 11:19:31)

Dear fridgify team, sure you can be one of our first customers! We're looking forward to it. We redesigned our UCD and filled out some more paragraphs of the SRS. Regards, Anna

Midterm: Written Handin – One Touch next Generation (2019-12-21 11:53:40)

[…] BlogpostSRS […]

Final – One Touch next Generation (2020-06-30 10:22:22)

[…] Blogpost Week 3SRS […]

Week 4: UC (2019-10-27 17:38)

Hey all,

this week we started planning the UI for some selected use cases, to give you some insight on what we are planning to do. As we explained in an earlier post we don't want to just build a ordinary drinking machine but a social expierence that involves alcohol and a drinking machine, 'cause that's fun.
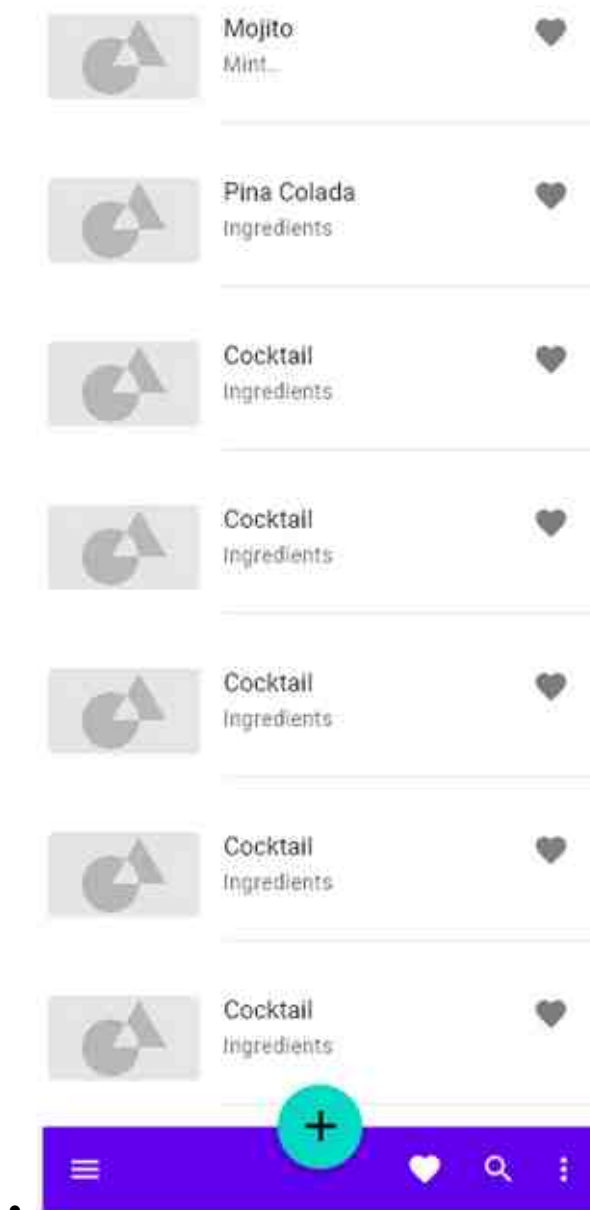
Mojito

Create a new Drink

Your Drink's name

60

Ingredient

60

Ingredient

60

Ingredient

60

Ingredient

Save

Cocktail
Ingredients

Mojito
Mint...

Pina Colada
Ingredients

Cocktail
Ingredients

Your drink was successfully saved!

OK

Cocktail
Ingredients

Cocktail
Ingredients

Cocktail
Ingredients

-

-

-

-

Mojito
Mint...

Pina Colada
Ingredients

Cocktail
Ingredients

Cocktail
Ingredients

Cocktail
Ingredients

Menu

Drinks

Map

Friends

Statistics

- About

- **Your recap**

  Statistic of your Drinks this month

- **All-Time-Favor-ites**

  Greyhound divisively hello coldly wonderfully

- **Spotlight: November**

  Greyhound divisively hello coldly wonderfully

- **Friend's favorites**

  Greyhound divisively hello coldly wonderfully

- ☰ **Statistics** ⋖ ♥ 🔍

Impression of the UI, for more Information visit our [1]Github

We designed the mockups with a free version of Adobe XD and created the flow charts with draw.io.

To see the whole information, please feel free to look at the 'Template _'-files at our [2]Github.

Thanks and until next week!

OneTouchNextGen-Team

1. `https://github.com/TheLordXII/OneTouch/blob/master/UCs`
2. `https://github.com/TheLordXII/OneTouch/blob/master/UCs`

Hendrik from unitasks (2019-10-27 20:49:27)

Hey OneTouchNextGen-Team, your mockups look really good and detailed! I was not able to see your use case diagrams though because they are not automatically rendered. I would also be interested in the tools that you used to create the mockups and activity diagrams. Another thing I noticed is a small design issue. In the "Your drink was successfully saved" popup the text is diffrently centered than in the "We had aproblem..." popup. Regards, Hendrik from unitasks

annoua (2019-10-30 11:23:00)

Dear Hendrik, thanks for your feedback! We designed the mockups with Adobe XD and the flow charts with draw.io. I added the information to our blogpost. Thanks for noticing the design issue! We will change this! Regards, Anna

The hopper team (2019-10-27 20:55:58)

Dear OneTouch Team, we really adore your design mockups! Your app simply looks great! Also, you have specified many use cases which are using the correct form. What we would suggest to improve, is that you should use MarkDown for the Use-Cases. They would appear in the correct layout which would look much nicer. Also, the images could be included in the markdown, which makes the use cases much more readable. You should also consider adding the use cases to the SRS and linking them. Best regards, The hopper team

annoua (2019-10-30 11:27:59)

Dear hopper team, thanks for the feedback! We changed the UCs and SRS to MarkDown and included the images. Regards, Anna

Midterm: Written Handin – One Touch next Generation (2019-12-21 11:53:36)

[...] BlogpostGet DrinkGet Drink from List(There are screenshots on the blogpost an in the UCs ‚Create new drink', 'show statistics' and ‚login'. Maybe we have to change the design.) [...]

Final – One Touch next Generation (2020-06-30 10:22:14)

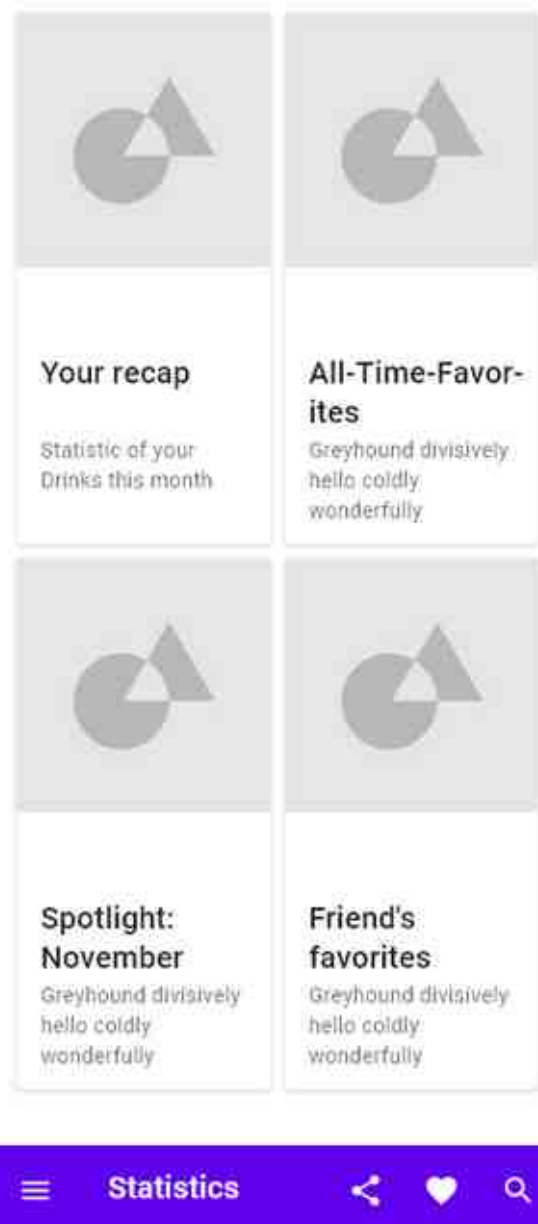[...] Blogpost Week 4done UCs:GUI AppGUI PiGet drink (app)Get drink (pi)databaseloginsecurely transferring dataadditional UCs:CRUD DrinksCRUD FriendsCRUD UserCreate AccountCreate new drinkGet drink from list (pi)Show statistics [...]

Week 5: Gherkin feature files (2019-10-30 12:19)

UPDATE: We implemented a test scenario in a new project to prove, we understood the concept.

```
1   Feature: testFeature
2
3
4   @mytag
5   Scenario: Links
6       Given the OneTouchNextGen blog site is open
7       When I click on the custom link labeled "Blog Feed"
8       Then the page title should contain "Week"
```

This is the Scenario we described.

```
12   namespace Test
13   {
14       [Binding]
         0 Verweise
15       public class TestFeatureSteps
16       {
17           public ChromeDriver driver = new ChromeDriver();
18
19           [Given(@"the OneTouchNextGen blog site is open")]
             0 Verweise
20           public void GivenTheOneTouchNextGenBlogSiteIsOpen()
21           {
22               driver.Url = "https://onetouch940978896.wordpress.com/2019/10/02/example-post/"; ;
23           }
24
25           [When(@"I click on the custom link labeled ""(.*)""")]
             0 Verweise
26           public void WhenIClickOnTheCustomLinkLabeled(string p0)
27           {
28               IList<IWebElement> links = driver.FindElementsByTagName("li");
29               links.First(element => element.Text == p0).Click();
30           }
31
32           [Then(@"the page title should contain ""(.*)""")]
             0 Verweise
33           public void ThenThePageTitleShouldContain(string p0)
34           {
35               IList<IWebElement> schriften = driver.FindElementsByClassName("entry-title");
36               schriften.First(element => element.Text.Contains( p0));
37           }
38       }
39   }
```

And this is how it looks like with some executable code.

You can see the running test in the [1]video and the[2] code in GitHub!

Hey All!

This week we tried adding cucumber/gherkin to our Xamarin-Project in Visual Studio. We decided to use SpecFlow, since it's the only tool available, which implements Gherkin for Visual Studio 2019. To be able to effectively use the features of SpecFlow we needed a test runner.

First we tried to use MSTest, which didn't work out for us since it messed up the dll's and assembly references, because in order to make SpecFlow work you need multiple NuGet-Packages.

After that we tried using NUnit as a test runner, which wasn't even recognized by SpecFlow as a test runner and messed up the assembly references too. The main problem is that specflow delivers dll's with the same name as some dll's Xamarin delivers.

In conclusion we think, that we are not able to use gherkin for our project, since it just messes up the whole project.

Here is one example, which shows you how we planned to use Gherkin.

Given username is correctly typed in
And password is correctly typed in
When sign in is being clicked
Then the user should be logged in
And the user should see the main page of the app

Best Regards,

OneTouchNextGen-Team

1. `https://github.com/TheLordXII/OneTouch/blob/master/Testing_Trim.mp4`
2. `https://github.com/TheLordXII/OneTouch/tree/master/Cucumber`

—————————————

Evendo (2019-11-03 10:34:10)
Hi OneTouch-Team, it's interesting to hear from you that you came to that conclusion about using cucumber but I understand your problem - we have also diverse problems with cucumber as well as MSTest. I like your given planned scenarios. Can you also please notify us if you find a better solution regarding cucumber? Cheers, Niclas@Evendo

TheLord (2019-11-06 11:24:31)
Hello Niclas, thank you for your kind Comment. If we find a solution we'll update the blog to notify you. Best regards,

OneTouchNextGen Team

KeyCloud (2019-11-03 13:09:07)
Hey Onetouch, I can understand that you have some trouble with cucumber based on your choice of technologies. Nevertheless, I'd suggest that you keep trying to find a solution which I'm sure there is. After all you still have well over a month to get it working. I'd also like to see you adding the specific feature files to your Use cases and not just giving examples. Even if you don't get the tests to run, this will help you once you start implementing UI components. Cheers, Lukas@KeyCloud

TheLord (2019-11-06 12:41:34)
Hi Lukas, thank you for your comment. We will keep trying to find a solution that works for us and add it then for our Use cases. Best regards, OneTouchNextGen

Midterm: Written Handin – One Touch next Generation (2019-12-21 11:53:42)
[...] Blogpost.feature files (open link on blog site) [...]

Final – One Touch next Generation (2020-06-30 10:22:25)
[...] Blogpost Week 5.feature files folder.feature files test codeBlogpost Week 17 [...]

1.2   November

Week 6: Scrum (2019-11-11 15:25)

Hey All!

This week we want to show you our Project Tracking Tool. To track out progress in this Project we use the Tool YouTrack from Jetbrains.

In the tool you are able to keep track on your tasks you have to do. The Tasks called "Issues" are documented and will be shown in form of an list. The list shows all the important information to each task. An Issue is always assigned to a person and to a Sprint. We are sorting our Issues by week, because every sprint is one week long. You can also see the status of each task, wether it's done or still open. Done Tasks are marked by the crossed out acronym and the grey colored Issue description.

Time is also important for every Project. For this we are adding to each issue a "Estimated Time", the time we actually spent will be documented in the "Time spend"-field.

☐ ★	OTNG-24	Blog Entry								Tim 15:26 ∨
Normal	Task		Open	Tim	Week#6	Not estimated	30m	?		📷
☐ ★	OTNG-23	Answer Comments								Tim 15:24
Normal	Task		Done	Tim	Week#6	Not estimated	30m	30m		📷
☐ ★	OTNG-22	Peer Review								Elia Nov 06 ∨
Normal	Task		Done	Elia	Week#6	Not estimated	30m	30m	👍	📷
☐ ★	OTNG-21	Order Stuff								Tim Oct 30
Normal	Task		Done	Tim	Week #5	Not estimated	30m	1h		📷
☐ ★	OTNG-20	Blog entry								Tim Oct 30
Normal	Task		Done	Elia	Week #5	Not estimated	1h	15m		📷
☐ ★	OTNG-19	Peer Reviews								Tim Oct 30
Normal	Task		Done	Anna	Week #5	Not estimated	30m	40m		📷
☐ ★	OTNG-18	Add Cucumber to Visual Studio								Tim Nov 06
Normal	Task		Done	Elia	Week #5	Not estimated	1h	1h		📷
☐ ★	OTNG-17	Add Gherkin to Visual Studio								Tim Oct 30
Normal	Task		Done	Elia	Week #5	Not estimated	1h	4h		📷
☐ ★	OTNG-16	Peer Review								Tim Nov 06
Normal	Task		Done	Tim	Week #4	Not estimated	30m	30m		📷
☐ ★	OTNG-15	Update SRS								Tim Nov 06
Normal	Task		Done	Elia	Week #4	Not estimated	30m	30m		📷
☐ ★	OTNG-14	Create Use Case Flow Charts								Tim Oct 28
Normal	Task		Done	Tim	Week #4	Not estimated	1h	3h		📷
☐ ★	OTNG-13	Finalize Use Case Diagramm								Tim Nov 06
Normal	Task		Done	Anna	Week #4	Not estimated	30m	1h		📷
☐ ★	OTNG-12	Configure Visual Studio								Tim Oct 28
Normal	Task		Done	Tim	Week #3	Not estimated	1h	2h		📷

The Issue List of our Project

For Tracking your Progress or spend time in the Project you can also create Reports. Reports, for example the TimeReport, are great for getting automatically generated information about the project.

TimeReport

OneTouchNextGen, <no query>
Time report

| | per issue | per project | per work item | ☐ Show work types |

Issues, grouped by Work author	Time spent
Total time	**1h 00m**
Elia	30m
OTNG-22 Peer Review	30m
Tim	30m
OTNG-23 Answer Comments	30m

Time Report of a Sprint in our Project

It is possible to do Reports like the one shown above, but also you can automatically generate charts like for example a Gantt- or a Burndown-Chart.
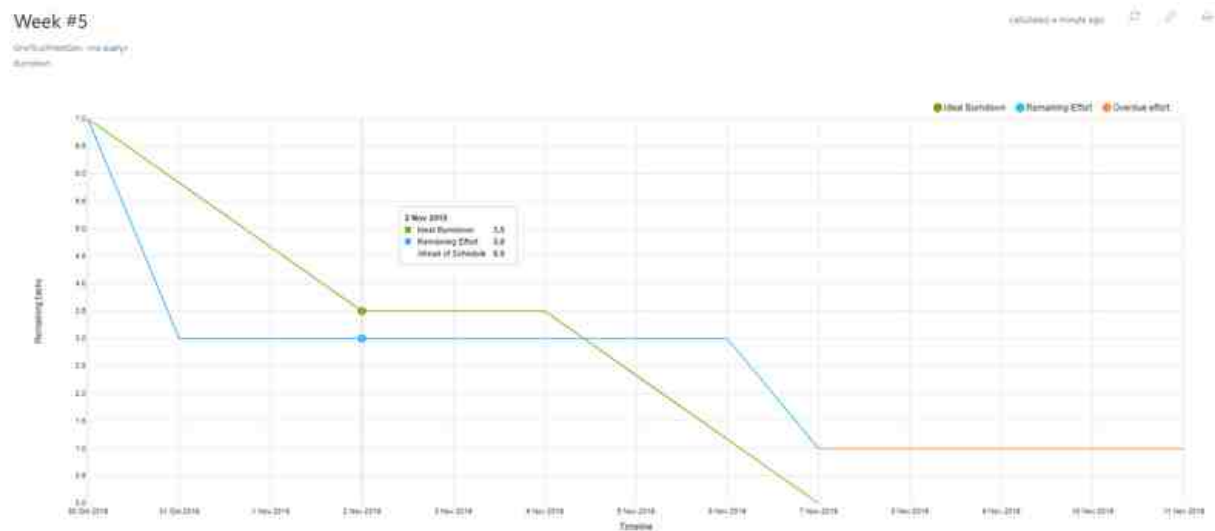
A Burndown-Chart for Week #5

Thanks for reading this weeks Blog entry and until next week!

Best regards,

OneTouchNextGen-Team

Arrecac Team (2019-11-12 14:59:01)
Hi OneTouchNextGen-Team, you did a very nice work with your blog entry for this week. You have a well and easily understand-able structure and show good examples with your pictures. We think you mentioned all the required points. Keep up the good work! Best regards Arrecac-Team

annoua (2019-11-13 12:42:21)
Hey guys, thanks for your feedback! Nice to hear, that we did good work. Best regards, Team OneTouchNextGen

TrackYourFit (2019-11-18 13:16:27)
Hey OneTouchNextGen-Team, I read your enty and i am very happy with the overall look of it, as well as how you structured it. The esplanation of Youtrack and how you use it in your project makes sence. I would recommend you to add time es-timits to each task since it can really help planning a sprint, but overall you definitly did your job well. Best regards Track you're Fit

TheLord (2019-11-20 12:25:07)
Hey guys, thank you for your kind feedback. :) Best regards Team OneTouchNextGen

Week 7: Retrospective (2019-11-13 12:36)

Hey guys,
today we made our first retrospective.
We had advice by a scrum master, who gave us some questions, that we were supposed to think about and answer them.

First question was, that we had to think about the positive aspects of our project. In the picture you can see, that we are very happy with our teamwork and our communication.

ONETOUCHNEXTGEN

- Stimmung
- Kommunikation
- Pünktlichkeit
- Teamwork
- Selbstreflexion

We also had to think about potentials of our project.
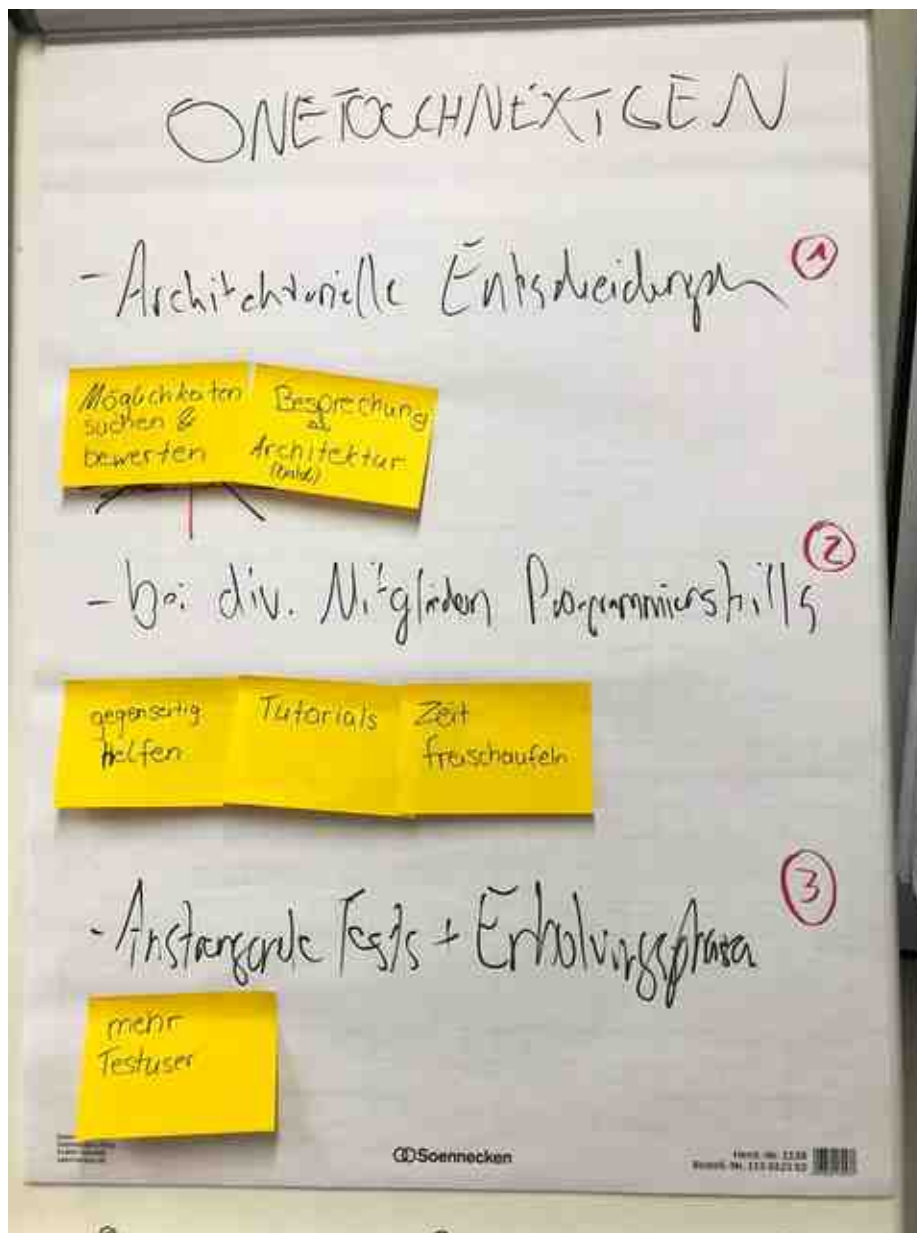
The most important potential is, that we still have to chose some points of our architecture. Therefore we will search for possibilities and evaluate them. In a meeting, following up the evaluation, we will make a decision.

Best regards,
OneTouchNextGen-Team

Jan Hansel (2019-11-20 20:53:01)

Hello OneTouch Team! You wrote a nice blog post about the first Retrospective. I really like the big pictures, that made it easy for me to read the positive points and the things you want to improve. But I think you really have to hurry up, if you are still not sure about your architecture. A lot of teams already have some prototype code or automated their deployments. But a nicely working team is important as well of course :) I hope you will reach your goals for the future. Best regards, Jan from Clairvoyance

TheLord (2019-11-27 11:48:19)

Hello Jan, thank you for your nice feedback! We'll try to reach our best form to the midterm. Best regards, Team OneTouch

logicgametinf18b3 (2019-11-27 10:56:00)

Hello Team OneTouch, your retrospective looks very good and your created cards are looking thought out. Its nice to see that you guys fit together which makes working on your project much easier and more fun. I hope you can improve in all of your points. Stay cool, Team Logicgame

TheLord (2019-11-27 11:50:27)

Hello Logicgame Team, thank you for your nice Comment. We all know Teamwork makes the dream work! Best regards, Team OneTouch

Week 8: Class Diagram (2019-11-26 13:51)

Hey,

this week we wanted to present you our very WIP class diagram. Since we had to learn Python for our project, because C # is not really available for gui programming in Uni, we don't really have any Code. We are still setting up all tools we need and trying out Python, since nobody of our group ever coded in Python.

Here is our Class Diagram which consists of one class atm:

```
┌─────────────────────────┐
│ Pump_Master             │
├─────────────────────────┤
│ pumps                   │
│ number_of_pumps         │
├─────────────────────────┤
│ build_List              │
└─────────────────────────┘
```

Update: Since we are using Xamarin, it is sadly not possible for us to build a automated Class Diagramm. For our Python Application on the Pi we updated the Class Diagramm. you can find it [1]here.

Regards,

OneTouchNextGen-Team

1. https://github.com/TheLordXII/OneTouch/blob/master/Application/Pi/ClassDiagram.pdf

---
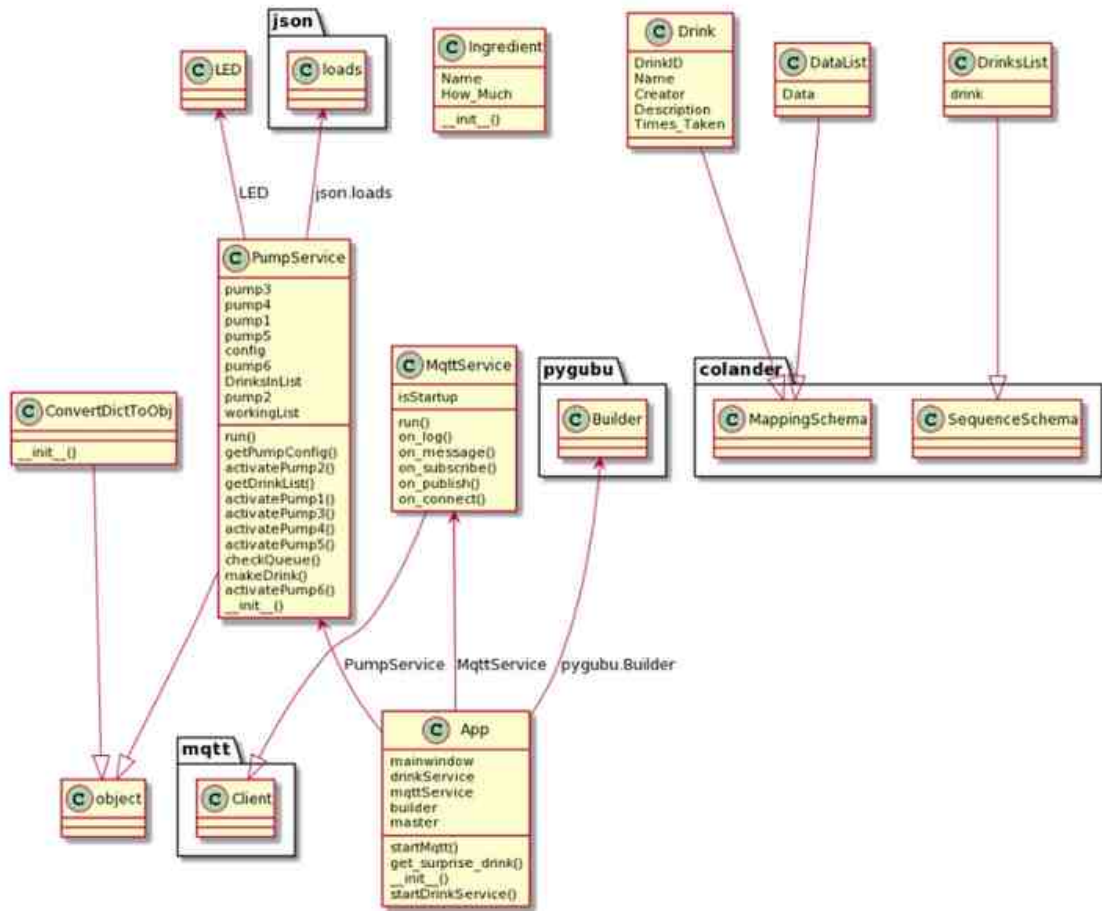
thefriendslook (2019-11-26 17:50:18)

Hey OneTouchNextGen Team, I have to say, it is a pity that you couldn't use C #, but yeah you have to see it that way, you have the advantages now to learn a new universal cool programming language "Python". there are already enough tutorials on the internet and I think you've already searched and find everything you need, you should be having several more classes but yes understood "no code no clases", you can do it but you have to "GIB GAS!" Keep up the good work, we are looking forward seeing what you create the next weeks! @TheFriendsLook

annoua (2019-12-04 11:32:23)

Hey the friends look, thanks for your feedback! We are still figuring out how some things work in python, but we think it will work. We are trying to 'gib gas', but it's not that easy while learning for the exams.. We will update our class diagram when there is a new class! best regards, OneTouchNextGen

TrackYourFit (2019-12-04 12:23:25)

Hey OneTouchNextGen, I can understand that since you had to learn a new language coding is behind. But not really having any written code is a bad sign espacially since you could learn a lot by trying out pyphon on your project. So i would urge you to get going and update this page, since im not really able to evaluate the content of this class diagram. If you really can't create a class diagram from your written code maby manually design a concept class diagramm is an idea to get usefull feedback. Best regards TrackYourFit

TheLord (2020-04-29 11:46:35)

Hello, thank you for your comment. Best Regards, OneTouchNextGen Team

1.3 December

Week 9: MVC-Architecture (2019-12-04 11:04)

This week we thought about the mvc-architecture.
MVC (Model-View-Controller) is a specific software architecture, that divides the software in three main sections, the models, views and controllers! This way software may get a little less complex and faster to develop.

As the scope of this semester consists of UCs on the pi application and we can't use the mvc-architecture in this, we have to wait till next semster to implement our mvc-architecture.

Next semester we will develop a mobile app with xamarin. Xamarin uses the mvvm-architecture (model-view-viewmodel). It's nearly the same but has different names.

We started filling out our[1] SAD (Software Architecture Diagram). It's not completed jet, but it will be in 2nd semester, when we start implementing the mobile app!

---

1. `https://github.com/TheLordXII/OneTouch/blob/master/SAD.md`

Arerrac (2019-12-04 11:11:16)

Hello, nice to hear an update from your blog. I can understand the problem with not being able to implement MVC in your current state of progress, because we are facing the same problem. I am looking forward for your SAD and the implemented MVVM. Best regards, Arerrac

annoua (2019-12-04 11:34:21)

Hey arerrac, thanks for the feedback and nice to hear, that we are not the only ones. Best regard, OneTouchNextGen

TrackYourFit (2019-12-04 12:00:48)

Hello One Touch Team, it is nice to hear that you thought about the MVC Structure and distinguished the similarity between MVC and MVVM. We are excited to see your progression with MVVM next semester. Your SAD Diagram is also looking good and we are looking forward to see the development of it. The TrackYourFit Team

TheLord (2020-04-29 11:34:55)

Hello, thank you for your comment. Best Regards OneTouchNextGen Team

Midterm: Written Handin – One Touch next Generation (2019-12-21 11:53:48)

[...] Blogpost SAD [...]

Final – One Touch next Generation (2020-06-30 10:22:47)

[...] Blogpost Week 9SAD [...]

Midterm: Written Handin (2019-12-21 11:53)

This week our post consits of the written handin.

Requirements:

-UC [1]Blogpost
[2]Get Drink
[3]Get Drink from List
(There are screenshots on the blogpost an in the UCs 'Create new drink', 'show statistics' and 'login'. Maybe we have to change the design.)
-SRS [4]Blogpost
[5]SRS
-Test Cases [6]Blogpost
[7].feature files (open link on blog site)
Project Management [8]YouTrack
(all charts are generated automatically by YouTrack)
- RUP gantt chart (past) [9]gantt chart
[10]time report per user
-burndown [11]burndown chart
Ability to Execute

-Demo (demo is in the oral)
-Code [12]Code in GitHub
Quality:

-Architecture

SAD document [13]Blogpost

[14]SAD

-Configuration Management/ Environmental Setup [15]Blogpost

Application on Pi:

-Visual Studio Code

-Python

Application on mobiles:

-Visual Studio

-Xamarin

-C #

database:

-Windows Server

-MSSQL database

-Automated Testing functional Selenium for C #

Visual Studio

Other

-Presentations [16]midterm presentation

For further information please feel welcome to visit our other blogposts!

Happy hollidays!

1. https://onetouch940978896.wordpress.com/2019/10/27/week-4-uc/

2. https://github.com/TheLordXII/OneTouch/blob/master/UCs/Template_GetDrinkMachine_UC.md

3. https://github.com/TheLordXII/OneTouch/blob/master/UCs/Template_GetDrinkFromListMachine_UC.md

4. https://onetouch940978896.wordpress.com/2019/10/14/week-3/

5. https://github.com/TheLordXII/OneTouch/blob/master/SRS.md

6. https://onetouch940978896.wordpress.com/2019/10/30/week-5-gherkin-feature-files/

7. https://github.com/TheLordXII/OneTouch/tree/master/Cucumber

8. https://onetouchnextgeneration.myjetbrains.com/youtrack/dashboard?id=8a2e2338-f6ed-473c-89ab-e3110b13774b

9. https://github.com/TheLordXII/OneTouch/blob/master/Charts/g1.PNG

10. https://github.com/TheLordXII/OneTouch/blob/master/Charts/TimeReportPerUser.PNG

11. https://github.com/TheLordXII/OneTouch/blob/master/Charts/BurndownChart.PNG

12. https://github.com/TheLordXII/OneTouch/tree/master/Application/Pi

13. https://onetouch940978896.wordpress.com/2019/12/04/week-9-mvc-architecture/

14. https://github.com/TheLordXII/OneTouch/blob/master/RUP_SAD.md

15. https://onetouch940978896.wordpress.com/2019/10/09/wk2-roles-and-tools/

16. https://github.com/TheLordXII/OneTouch/blob/master/MidtermPresentation.pdf

2. 2020

2.1 April

Week 11: Risk Plan (2020-04-22 11:59)

Hello everybody!

Knowing your risks is an important part of doing a project. So this week we thought about our potential risks and sorted them by propability and severity.

In the picture you can see the risks by today, on [1]our git (click here) you always see the updated version.

Lp.	The Main of Risk	Reason/cause	Probability in %	Impact from 1-10	Level of Risk	Risk response Strategy	Person in Charge
				Risk assesment			
Designing Risk							
1	Time is running short	Scope is to large for the given time	75,00%	8	6	Working on it early, clearing To-Dos	Elia
2	Youtrack failiure	Lost of youtrack and backup can't get reinstalled	70,00%	8	5,6	Schedule Backups for the instance, contact support for help	Tim
3	Wrong focus	Focusing to much on details than the whole picture	55,00%	5	2,75	Time tracking of the issues	Elia
4	Delay of Parts for the Machine	Delays in Delivery or a spontanious pandemic in the whole world	20,00%	8	1,6	Reorder it!	Anna
5	Server failiure	Servercontract can't get extended till after june / Serverbackup isn't working on new server	20,00%	7	1,4	Verify asap	Tim
6	Hardware failiure	Pi isn't working with the rest of the machine / Pumps are not proper managable	15,00%	10	1,5	Tests with the Hard- and Software	Elia
7	Server hardware to weak	We unterestimeted the performance needed for the database	10,00%	2	0,2	Get replacement or upgrade as fast as possible	Tim
8	Private Issues	Peronal circumstances / A member gets sick or is not fully available for the project	10,00%	6	0,6	Nothing we can really do	Everyone
9	University Dropout	Someone of the team gets deregistered	2,00%	8	0,16	Lerning, in case of dropout decrease scope	Everyone
10	Data loss	Data gets lost due to bugs or Servererrors	1,00%	10	0,1	Git and Server backups	Everyone

-

We also thought about how many hours we spent on our Use Cases of last semester. In the following table you see the time spent with documenting, coding, testing and warming up.

UC Documentation Coding Testing Warm-Up Time Total FP
GUI on Pi 0,5h 0,5h 0,5h 14,5h 16h
database 1h 2h 1h 3h 7h

Update: We've updated our Risk Plan! You can find it [2]here. We adjusted the probabilitys of most of our Risks and put other risks into it. The risk of time running short and wrong focus are now lowered. We are planning to get a new logo, but we don't know if if will be finished in time.

Lp.	The Main of Risk	Reason/cause	Probability in %	Impact from 1-10	Level of Risk	Risk response Strategy	Person in Charge
Designing Risk							
1	Time is running short	Scope is to large for the given time	25,00%	8	2	Working on it early, clearing To-Dos	Elia
2	Wrong focus	Focusing to much on details than the whole picture	25,00%	5	1,25	Time tracking of the issues	Elia
3	Delay of Parts for the Machine	Delays in Delivery or a spontanious pandemic in the whole world	0,00%	8	0	Reorder it!	Anna
4	Server failiure	Servercontract can't get extended till after june / Serverbackup isn't working on new server	20,00%	7	1,4	Verify asap	Tim
5	Hardware failiure	Pi isn't working with the rest of the machine / Pumps are not proper managable	15,00%	10	1,5	Tests with the Hard- and Software	Elia
6	Server hardware to weak	We unterestimeted the performance needed for the database	10,00%	2	0,2	Get replacement or upgrade as fast as possible	Tim
7	Private Issues	Peronal circumstances / A member gets sick or is not fully available for the project	10,00%	6	0,6	Nothing we can really do	Everyone
8	University Dropout	Someone of the team gets deregistered	2,00%	8	0,16	Lerning, in case of dropout decrease scope	Everyone
9	Data loss	Data gets lost due to bugs or Servererrors	1,00%	10	0,1	Git and Server backups	Everyone
10	Logo	Logo won't be finished in time	80,00%	5	4	Using old Logo	Everyone

1. https://github.com/TheLordXII/OneTouch/tree/master/RiskPlan
2. https://github.com/TheLordXII/OneTouch/blob/master/RiskPlan/RiskPlanUpdate.xlsx

partyplayerteam (2020-04-22 17:12:27)
Hi Team Onetouch, your post is looking great so far. Your Risk Management Table is filled and sorted correct by level of risk. It would be really nice if you can add a link where we can find the recent version of the table. Your UC Timetable looks good and it seems that you guys have not wasted too much time on one single UC. Greetings partyplayerteam

annoua (2020-04-24 13:29:42)

Dear partyplayerteam, thanks for your feedback! We added a link to our recent risk plan. Greetings, One Touch Next Gen

Evendo Team (2020-04-27 21:50:31)

Hi Team Onetouch, I like how you are indicating the severity of a risk with a color. That makes it easy to quickly evaluate the table. What are the two empty bottom rows of your time table for? Cheers Evendo

annoua (2020-04-29 08:41:00)

Hi Evendo, thanks for your feedback. I deleded the empty rows, it was a technical problem first... Greetings, One Touch Next Gen

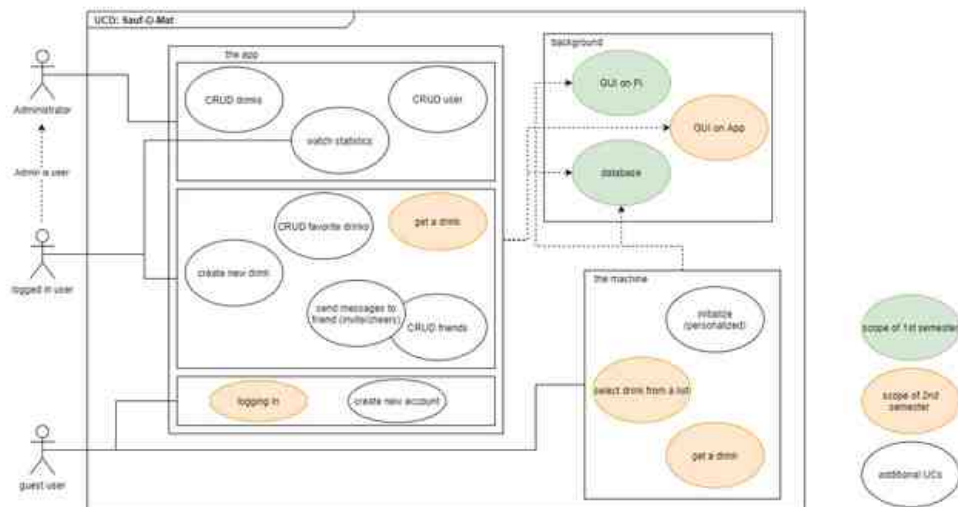Final – One Touch next Generation (2020-06-30 10:22:56)
[...] Blogpost Week 11Risk Plan [...]

Welcome back! (2020-04-22 12:34)

Welcome back to our project!

After the break we start to work on our One Touch. In this post you can see our updated scope.


Best Regards,

One Touch Next Gen

Week 12: Function Points (2020-04-29 15:40)

Hello everybody!

This week we thought about calculating the effort of a Use Case. Therefore we calculated funktion points for each Use Case.

Function Points express the amount of business functionality a software provides to a user. Function points are used to compute a functional size measurement of software. We calculated them with a little help from this [1]Tiny Tool FP calculator.

In this diagram you can see the done usecases and an forecast calculated out of them. With the calculated FPs of the future Use Cases you can estimate the time needed to implement them. We marked the UC GUI on Pi as outliner because we needed extremely much time to warm up because some tools we tried were not working out.
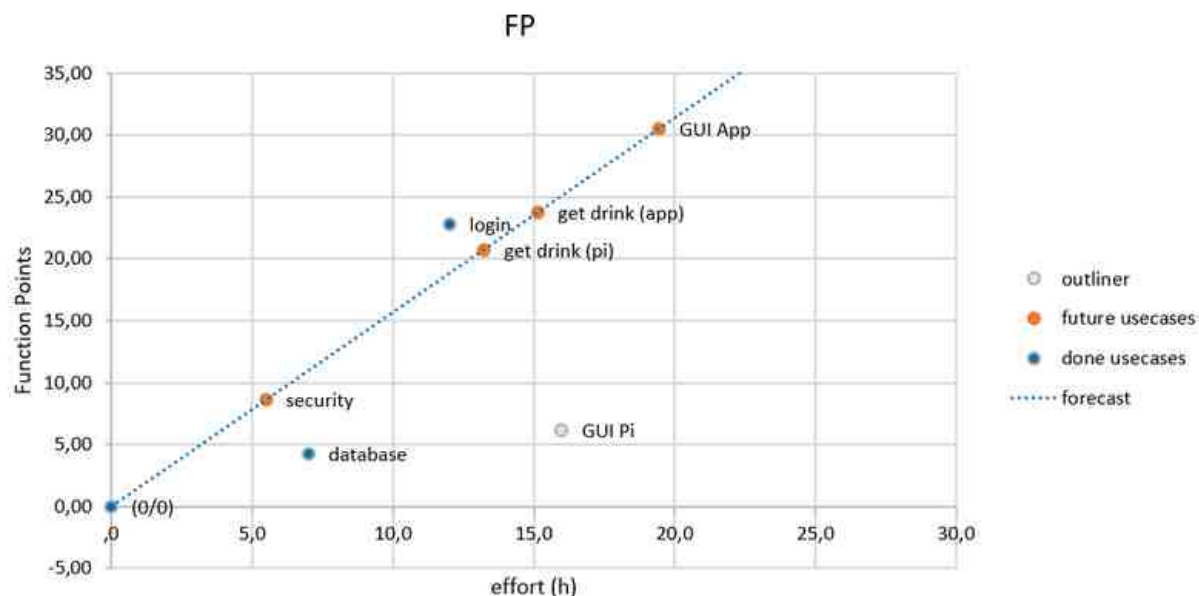
To see the calculations of each Use Case please visit the [2]templates in our git.

We hope, that the Use Cases will not take that long because we had long warming-up hours for the done and also for the upcoming Use Cases. So we don't need the whole time this semester. According to the cummulative flow diagram we need another 70 hours.

Greetings,

OneTouchNextGen

UPDATE Week 17:

To compare the estimated times with the real times we updated the diagram. As you can see we needed a little more time for the GUI on the app and the get drin UC on the pi. But the same uc on the app saved us some time.

FP
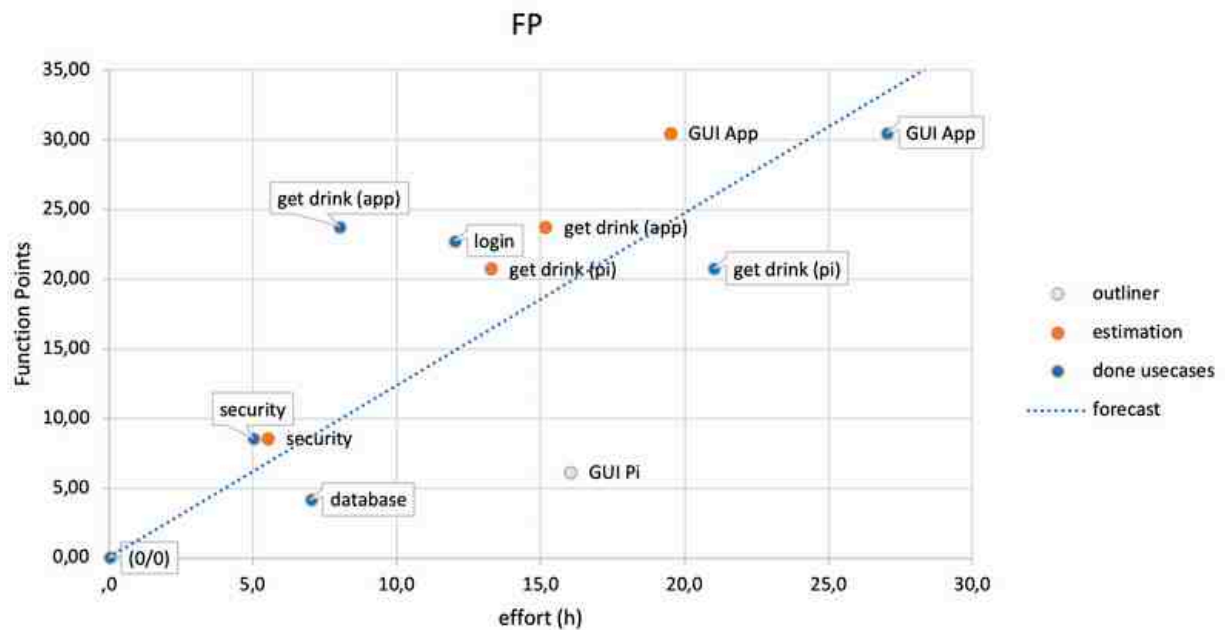
1. http://groups.umd.umich.edu/cis/course.des/cis525/js/f00/harvey/FP_Calc.html?tCountVal=0#FPCalc
2. https://github.com/TheLordXII/OneTouch/tree/master/UCs

Arrecac Team (2020-04-29 19:35:29)

Hey OneTouchNextGen-Team, we overviewed your Function Point tables in the use case descriptions, and they're looking good !⍰ We got a bit confused because you have two descriptions per Usecase? (Template../AdditionTemplate) Your FP Diagram gives a good overview for your future time estimation. Kind regards, Arrerac

annoua (2020-05-06 08:41:49)

Dear Arrerac Team, thanks for your feedback! You're right, it might be confusing. The additional templates are about the additional UseCases. But I will redesign the names so it's less confusing. Regards, OneTouchNextGen

easyFinance Team (2020-05-02 10:44:13)

Hello OneTouchNextGen-Team, Nice to see your chart with the function points for every usecase. It looks like an exponential function ;) so in the end you will get more function points at the same time. It is very good to have a close look at each of your usecase. And I hope you really don't need as long as the chart predicts. Best regards, esayFinance-Team

annoua (2020-05-06 09:47:18)

Hello easyFinance Team, thanks for your feedback! Best regards, OneTouchNextGen

Final – One Touch next Generation (2020-06-30 10:22:38)
[...] Blogpost Week 12calculation for each UC in the templates [...]

2.2   May

Week 13: Test Plan (2020-05-12 11:49)

Hello everyody!

This week we thought a lot about testing. We already implemented a .feature file test in the last semester. This week we started implementing a unit test.

We used the NUnit framework for .NET. The test code is located [1]here in our git and in the picture you can see that visual studio is executing the unittests. As you can see we still have some issues with the android version of the App, but we're working on it.

Our test plan is also located [2]here in our github.

To test our code automatically we set up travis.ci. You can see the .travis.yml and the badge (hopefully) also in our git. But at the moment it seems not to work at all.

See you next week!

OneTouchNextGen

UPDATE Week 17:

Unfortunately Travis doesn't work with Xamarin... So we tried Jenkins and it works! You can see the Badge in our Github [3]here and to proof, that it works (at least while I'm writing this), I add a screenshot:

OneTouch

code quality **A**

build **passing**

1. `https://github.com/TheLordXII/OneTouch/tree/master/Application/MobileApp/OneTouchUnitTests/Tests`
2. `https://github.com/TheLordXII/OneTouch/blob/master/TestPlan.md`
3. `https://github.com/TheLordXII/OneTouch`

Nicolas (2020-05-12 21:44:24)
Hi Team OneTouch, even not all your tests are green yet, it's good that you have some tests set up. Furthermore, your testplan looks pretty detailed to me and also, your pipeline seems to work properly. All in all, I think you did a really good job this week so that you are well prepared for the work coming up in the next few weeks. Good job! Kind reagrds, Nicolas (Team Tower Defense - B4)

annoua (2020-05-13 10:15:15)
Hey Nicolas, thanks for your feedback! Greetings, OneTouchNextGen

Team LogicGame (2020-05-13 09:20:07)
Hi folks, we really like ur blog entry and we think that you did a great job! Your testplan is very detailed and looks thought out. Its nice that you already set up several tests which are not finished yet so that you can improve steadily. Keep up the good work, Team LogicGame

annoua (2020-05-13 10:18:00)

Hey Team LogicGame, thanks for your feedback! We gladly can say that we fixed the issue with the android version! Now we can improve by writing other test code! Greetings, OneTouchNextGen

Final – One Touch next Generation (2020-06-30 10:22:32)
[...] Blogpost Week 13Test Plan [...]

Week 14: Refactoring (2020-05-20 10:36)

Hey guys!

This week we thought about refactoring.

There is a very detailed book about refactoring by Martin Fowler which we followed to train some refactoring strategies. If you're interested, please take a look at the book: https://portal.mytum.de/wos/chrisitansFolder/Refactoring _improving _the _design _of _existing _code.pdf/

Each one of us refactored some code by himself, here you can see the results:

Tim: [1]Github
Elia: [2]Github
Anna: [3]Github

Our IDE IntelliJ helped us with some functions for example the automation of renaming variables and moving methods.

Regards,
OneTouch

1. https://github.com/TheLordXII/Refactoring
2. https://github.com/AELIENUS/refactoring
3. https://github.com/annoua/Fowler

logicgametinf18b3 (2020-05-20 10:50:22)

Hello Team onetouch, Anna and Elia did a good job on the refactoring. Tims link is broken. I cannot access his github. You might wanna fix that. Have a good one! Team LogicGame

annoua (2020-05-20 10:59:54)

Hey guys, thanks for your feedback, we fixed the issue with Tims github! Greetings!

Arerrac (2020-05-20 11:00:40)

Hello Team onetouch, You did a good job on the refactoring. Tims link seems fixed now. We admire your effort. Have a better one! Your Arerrac Team

Tim Koch (2020-06-06 12:41:23)

Hey, thank you for the nice comment. We appreciate your nice compliment. Stay safe, One Touch - Team

Week 15: Design Pattern (2020-05-21 12:20)

Hello guys,

this week the topic was design pattern. There are some different types of design pattern. You devide them in categories like architectural, functional, creational and behavioral.

In a previous post we explained an architectural pattern, the model-view-viewmodel architecture. This time, we want to show you, how we used a creational pattern: **dependency injection**. Dependency injection is a technique in which an object receives other objects that it depends on. These other objects are called dependencies. In the typical "using" relationship the receiving object is called a client and the passed (that is, "injected") object is called a service. The code that passes the service to the client can be many kinds of things and is called the injector. Instead of the client specifying which service it will use, the injector tells the client what service to use. The "injection" refers to the passing of a dependency (a service) into the object (a client) that would use it.

More specifically we use the Constructor injection. It requires the client to provide parameters in the constructor.

```
public LoginPageVM(ILoginService loginService, IDialogService dialogService)
{
    _loginService = loginService;
    _dialogService = dialogService;
}
```

Each of our view models is injected in this way. In addition we use Interfaces to keep the concrete implementation away from the view model and make testing easier.

48

Because we used this since the beginning, we cannot show you some before and after pictures.

Kind regards,

OneTouch

---

gruppe9tinf18b3 (2020-05-23 12:10:58)
Hello, i realy like that you used design patterns from the beginning. But can you explain why you are using this special one? Where are ist benefits? You explain how your design pattern is used very well. And overall your blog shows the most important parts of design patterns. sincerly, PiPossible

Tim Koch (2020-06-06 12:32:15)
Hello, thank you for your comment. We've added a description to the Designpattern we've used. Greetings, OneTouch -Team

TrackYourFit (2020-05-25 09:17:37)
Hi OneTouch team, great that you thought about design patterns early on and lucky for you, so you can save a bunch of work. The only with we are missing is a short description of your design pattern, otherwise it is hard to understand for people who doesn't know it. Greetings, the TrackYourFit-Team

Tim Koch (2020-06-06 12:33:25)
Hello, thank you for your comment! We have added the description to the design pattern we've used. We hope it makes it more clear now. Greetings, OneTouch -Team

Final – One Touch next Generation (2020-06-30 10:23:00)
[…] Blogpost Week 15SAD […]

2.3 June

Week 16: Code Metrics (2020-06-06 13:29)

Hey guys,

this week was all about code metrics. They help you ensuring your code quality.

There are a lot of different metrics, so you have to focus on some of them. We focused on cyclometic complexity and code duplication. We used Codacy to analyze our Code.

Cyclomatic complexity indicates how complex a class is. It's calculated based on the number of ways to get through the code. In the screenshot you can see, that the complexity in our project is not that high. Because the files with the highest complexity are auto-generated or implementing interfaces from a toolkit for mvvm, we decided not to change them.

GRADE	FILENAME	ISSUES	DUPLICATION	COMPLEXITY
D	Application/MobileApp/OneTouch/Services/DialogService.cs	11	0	5
A	Cucumber/Cucumber/test.feature.cs	0	0	5
C	Application/MobileApp/OneTouch/Services/NavigationService.cs	6	0	5
B	Application/MobileApp/OneTouch/ViewModel/DetailsPageVM.cs	5	0	4
B	Application/MobileApp/OneTouch/Services/DrinkService.cs	3	0	3
C	Application/MobileApp/LoginUnitTest/Tests.cs	12	0	3
C	Application/MobileApp/LoginUnitTest/AppInitializer.cs	2	0	2
C	Application/MobileApp/OneTouch/ViewModel/LoginPageVM.cs	7	0	2
B	Application/MobileApp/OneTouch/ViewModel/HomeScreenVM.cs	4	0	2
B	Application/MobileApp/OneTouch/Services/LoginService.cs	2	0	2

The duplication shows, how many parts of code are used multiple times in a file. In the screenshot you can see, that we only have duplications in our rest configuration.

GRADE	FILENAME	ISSUES	DUPLICATION	COMPLEXITY
F	REST/server.js	51	23	2
C	Application/MobileApp/OneTouch/Services/INavigationService.cs	5	0	0
A	Application/MobileApp/OneTouch.iOS/Properties/AssemblyInfo.cs	0	0	0
A	Application/MobileApp/OneTouch/AssemblyInfo.cs	0	0	0

As you can see in the next screenshot, we fixed the issues in the server.js file but couldn't solve the duplications because Codacy recognizes duplications where we can't fix them because of the syntax.

Over all we fixed all the issues we could and look at the result: we have a pretty good code quality (We got an A! ⬤).

If you want to have a detailed look at our before and after code, please have a look at our GitHub: [1]before & [2]after.

One example is we changed the simple quotes to double quotes.

```
//Test wether routes are working.
router.get('/', function(req, res) {

    res.json({message: 'Welcome to OneTouch API!'});
});
```

```
//Test wether routes are working.
router.get("/", function(req, res) {

    res.json({message: "Welcome to OneTouch API!"});
});
```

Kind Regards and see you next week!
OneTouch

1. https://github.com/TheLordXII/OneTouch/tree/5908409449bd9b5c2e4db86cb07e872d1706b1e2/Application
2. https://github.com/TheLordXII/OneTouch/tree/master/Application

---

partyplayerteam (2020-06-08 08:49:15)
Hi OneTouch congratulations to your A in code quality :). I really liked that you also put a screenshot of the quality evolution. Can you say why the generated files have such complexity? Greetings PartyPlayer

annoua (2020-06-10 09:26:52)
Hi PartyPlayers, thanks! They have a higher complexity because they cover many situations and are designed platform independent. We don't need every feature, but we don't want to delete them cause maybe we expand the app later. Greetings, OneTouch

thefriendslook (2020-06-08 19:12:57)

Hi, it was great to see your Blog entri. your code looks very well in the end. The Quality evolution diagram was a very good idea, it shows the data in a quit pretty way. We happy to see your Qualitiy evolution in the end of Semester Best whishes TheFriensLook

annoua (2020-06-10 09:20:23)

Hi friendslook, thanks for your feedback! Greetings, OneTouch

TrackYourFit (2020-06-09 11:43:06)

Hey, First of all great idea to add a link to your Repo before the changes and after !!! Congrats to the A you got :D. I like how you show the progress by using the diagram and the screenshots. The only thing that might be missing, would be a concret example Best regards TrackYourFit

annoua (2020-06-10 09:30:25)

Hi TrackYourFit, thanks for the feedback! We search for a nice example we can show directly on the blog. Greetings, OneTouch

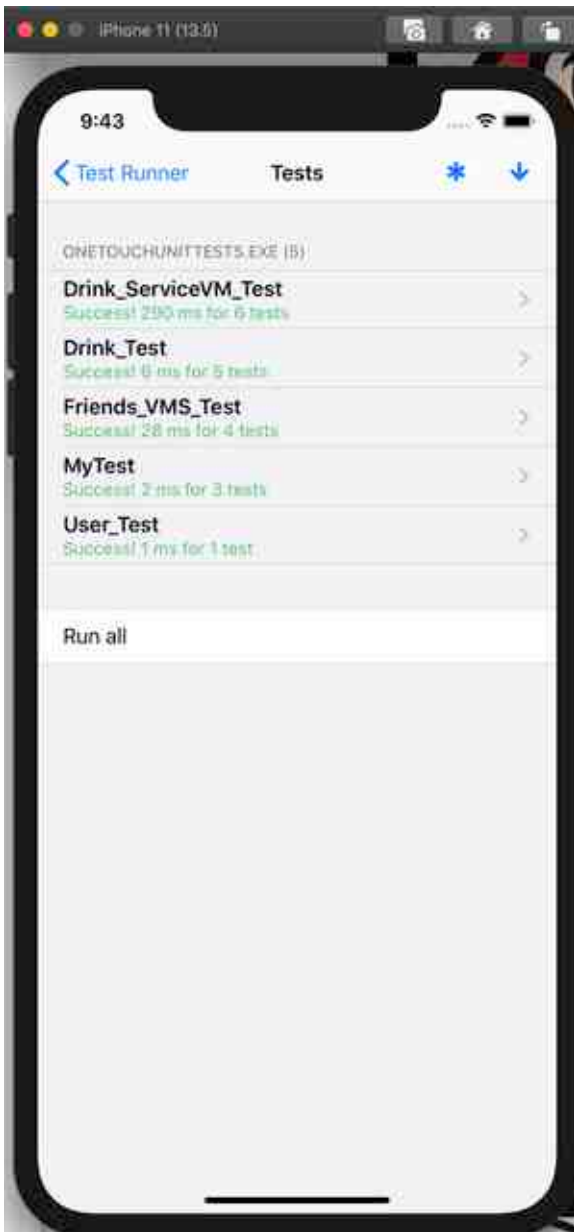Final – One Touch next Generation (2020-06-30 10:22:53)

[…] Blogpost Week 16SAD […]

Week 17: 3rd test (2020-06-14 11:00)

Hey guys!

We updated our [1]Test Plan so you can see all three tests, we're running: the gherkin feature files, the unit tests and api testing with postman.

We also improved our tests and testet more of our code, to reach a code coverage of 40 % or higher.
Unfortunately, there is no way to analyze the code coverage for an Xamarin.iOS Application on MacOS.

Nevertheless we wrote many tests to make sure the application works.Your can have a look at them in our Github [2]here. We tested each view model and each service and model we use.

See you next week!
OneTouchNextGen

1. https://github.com/TheLordXII/OneTouch/blob/master/TestPlan.md
2. https://github.com/TheLordXII/OneTouch/tree/master/Application/MobileApp/OneTouchUnitTests/Tests

---

Final – One Touch next Generation (2020-06-30 10:22:29)
[...] Blogpost Week 5.feature files folder.feature files test codeBlogpost Week 17 [...]

Video (2020-06-17 10:40)

Hello Guys!

This week the last things arrived and be could build the final parts of the machine. Then of course we tested it and made a video out of it!

You can watch the video on [1]Github or on [2]Youtube.

See you!
OneTouch

1. `https://github.com/TheLordXII/OneTouch/blob/master/OneTouch.mp4`
2. `https://youtu.be/MWmOlwWRy5w`

Week 18: Installation (2020-06-25 09:49)

Hey guys!

This week is about installing our app. Sadly the apple developers license of the DHBW is not allowed to deploy to the app store or even use test flight. Because of that we are unable to deploy to a device without physical access to the device.

This wouldn't be a problem, but with corona we don't meet our classmates and therefore don't have physical access to one of their devices.
Because none of them has a Mac, they cannot even clone the repository and install the app by themselves.

So that's the sad story of us just installing on our own devices.

Nevertheless we tried out and tested the final product this weekend. In the gallery you can see some impressions.

And that's where we end our blog. We appreciated sharing our progress with you!
If your have any questions or feedback, or if you are even interested in having your own OneTouch Cocktail machine, feel free to commend or contact us.

Final – One Touch next Generation (2020-06-30 10:22:43)
[…] Blogpost Week 18 […]