

lessons learned
Practices and Principles
of Software Design



2021

@majcon



감사합니다
GRAZIE VINAKA
TERIMA KASIH
THANK YOU
TAKK
merci
ありがとう
謝謝

for inviting me



for training me



Fun
Wireless
Well
to
Fish
Top
the
Fail
Federation
The
Weekly
World
Five
Fly
Takedown
To
Forum
Where's
What
Funny
Food
Way
Wrestling
That's
Technology
Tennis
Focus
Time
OK

```

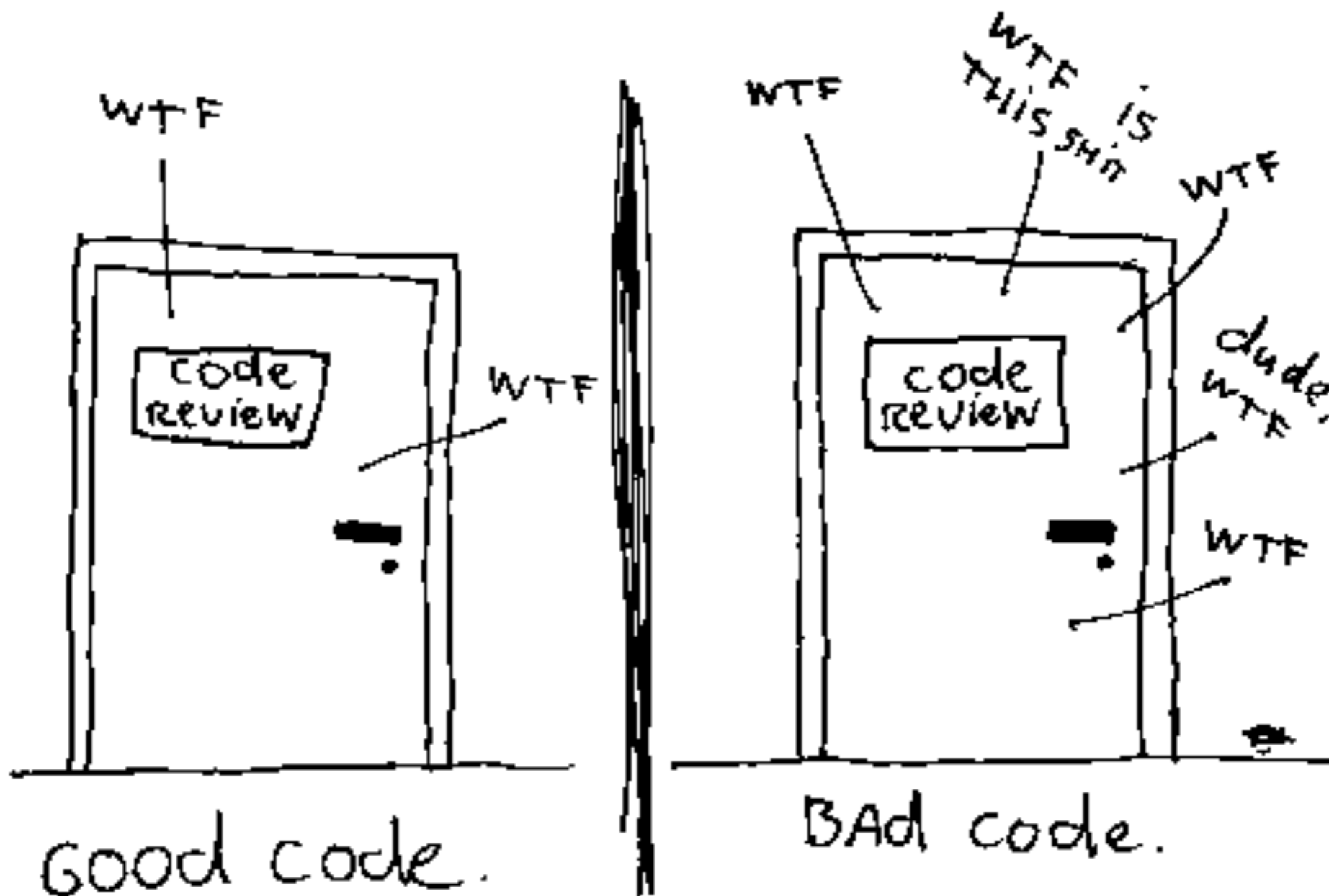
$(document).on('click', '.check_district_li', function() {
    if ($('#input', this).is(":checked")) {
        var old_text = $(".choose_city_text").text();
        var remove_text = $(".check_district_lable", this).text();
        var new_text = old_text.replace(", " + remove_text, "");
        $(".choose_city_text").text(new_text);
        var deselected = $('#input', this).val();
        var selected_var = $("#division_id").val().toString();
        var new_string = selected_var.replace(deselected + ',', '');
        $('#division_id').val(new_string);
        $(".check_district_lable", this).css("color", "#9da9be");
        $(".customCheckbox", this).removeClass('customCheckboxChecked');
        $('#input', this).prop('checked', false);
        $(".loaded_sectors").load("/ajax/load_sectors_bubble.php?district=" + new_string, function() {
            var selected_sectors = $("#sector_id").val();
            var result = selected_sectors.split(',');
            for (i = 0; i < result.length; ++i) {
                $(".loaded_sectors_span_" + result[i]).addClass('sector_checked');
                $('img', ".loaded_sectors_span_" + result[i]).attr("src", "/images/sector_bubble_close.png");
            }
            var sectors = [];
            $.each($(".sector_checked"), function() {
                sectors.push($(this).attr('id'));
            });
            console.log(sectors);
            var sectro_vars = $("#sector_id").val();
            var new_sectors = sectro_vars.replace(sectors + ',', '');
            $("#sector_id").val(new_sectors);
        });
    } else {
        var id = $('#input', this).val();
        var old_text = $(".choose_city_text").text();
        var new_text = old_text + ", " + $(".check_district_lable", this).text();
        $(".choose_city_text").text(new_text);
        var selected_var = $("#division_id").val();
        var my_val = selected_var + id + ",";
        $('#division_id').val(my_val);
        $(".check_district_lable", this).css("color", "#363e4a");
        $(".customCheckbox", this).addClass('customCheckboxChecked');
        $('#input', this).prop('checked', true);
        $(".loaded_sectors").load("/ajax/load_sectors_bubble.php?district=" + my_val, function() {
            var selected_sectors = $("#sector_id").val();
            var result = selected_sectors.split(',');
            for (i = 0; i < result.length; ++i) {
                $(".loaded_sectors_span_" + result[i]).addClass('sector_checked');
                $('img', ".loaded_sectors_span_" + result[i]).attr("src", "/images/sector_bubble_close.png");
            }
        });
    }
});
});
});

```



WTF!?

The ONLY VALID MEASUREMENT
OF CODE QUALITY: WTFs/MINUTE



```
public function check_login()
```

```
{
```

```
    if (!$SESSION['uid']){
```

```
        redirect( uri: "login");
```

```
    }
```

```
}
```

```
public function isLoggedIn()
```

```
{
```

```
    if (isset($SESSION['uid']) && !empty($SESSION['uid'])){
```

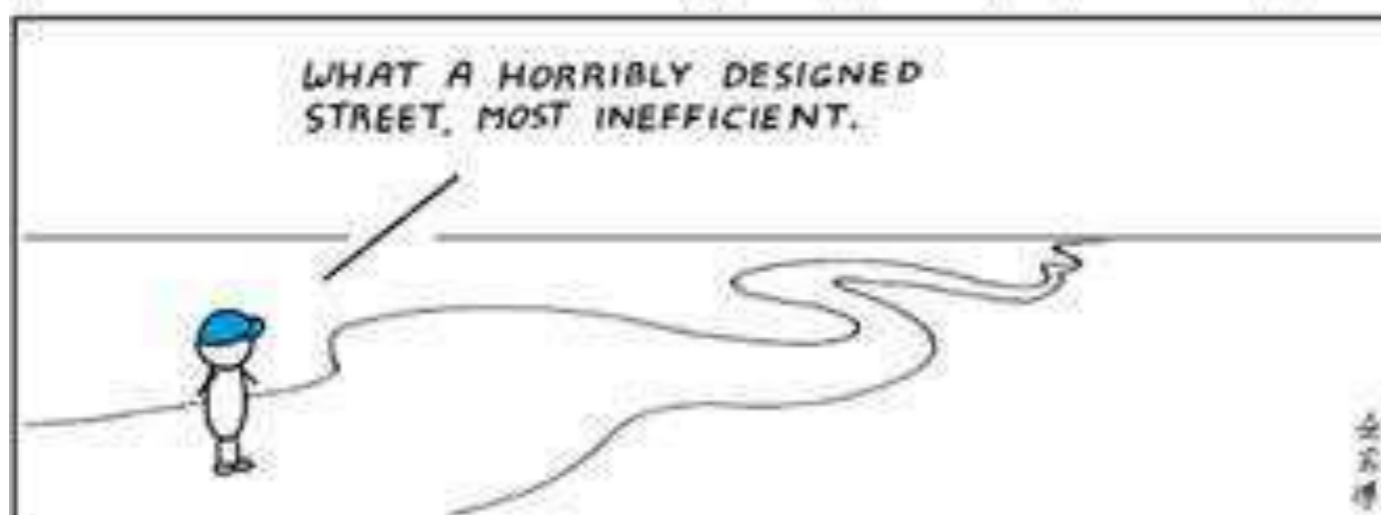
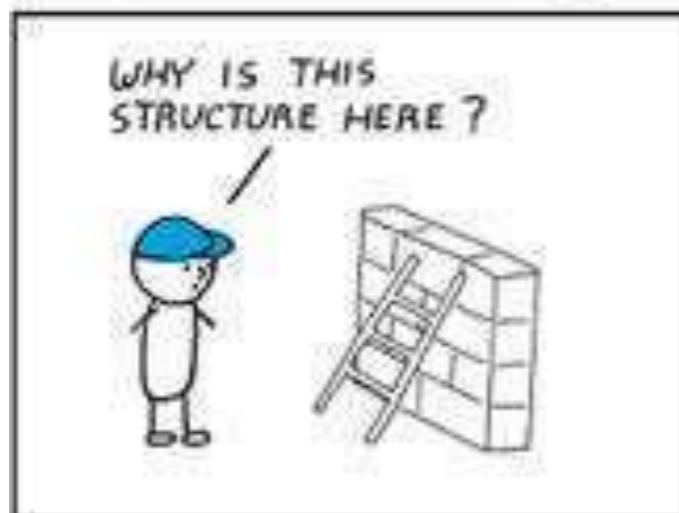
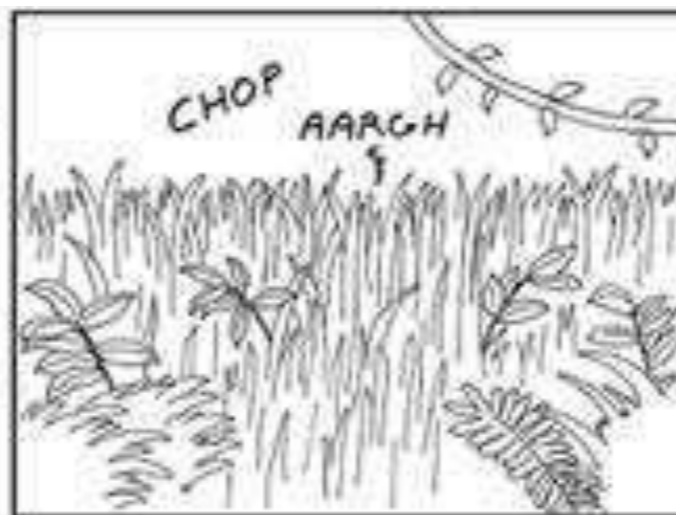
```
        return true;
```

```
    }
```

```
    return false;
```

```
}
```

WTF!?



I hate reading other people's code.


```

while(!reader.atEnd())
{
    reader.readNext();
    if(!reader.isStartElement() && reader.name() != "PropertySetDef")
    {
        while(!(!reader.isEndElement() && (!reader.name() == "PropertySetDef"),))
        {
            reader.readNext();
            if(!reader.isStartElement())
            {
                auto elementName = reader.name();

                if(!elementName == "Name")
                    propertySetName = reader.readElementText().trimmed();

                else if(elementName == "ApplicationClasses")
                {
                    while(reader.readNextStartElement())
                    {
                        if(reader.name() == "ClassName")
                        {
                            auto className = reader.readElementText().trimmed();
                            if(!className.isEmpty())
                                classList << className;
                        }
                    }
                }
                else if(elementName == "PropertyDefs")
                {
                    while(!(!reader.isEndElement() && reader.name() == "PropertyDefs"))
                    {
                        reader.readNext();
                        if(!reader.isStartElement() && reader.name() == "PropertyDef")
                        {
                            QJsonObject property;
                            while(!(!reader.isEndElement() && reader.name() == "PropertyDef"))
                            {
                                reader.readNext();
                                if(!reader.isStartElement())
                                {
                                    if(reader.name() == "Name")
                                        property.insert("name", reader.readElementText().trimmed());

                                    else if(reader.name() == "PropertyType")
                                    {
                                        while(!(!reader.isEndElement() && reader.name() == "PropertyType"))
                                        {
                                            reader.readNext();
                                            if(!reader.isStartElement())
                                            {
                                                auto valueType = reader.name().trimmed();
                                                if(!valueType == "TypePropertySingleValue")
                                                {
                                                    while(reader.readNextStartElement())
                                                    {
                                                        if(reader.name() == "DataType")
                                                        {
                                                            QString ifcValueType = reader.attributes().value("type").toString().trimmed();
                                                            if(!ifcValueType.isEmpty())
                                                            {
                                                                if(!ifcValueType == "IfLogical")
                                                                {
                                                                    property.insert("typeifc", "IfLogical");
                                                                    property.insert("type", "enum");
                                                                    property.insert("usertype", "");
                                                                    QJsonObject typeSpec;
                                                                    typeSpec.insert("enumname", "PEnum_Logical");
                                                                    typeSpec.insert("enumitems", "true,false,unknown");
                                                                    property.insert("typespec", typeSpec);
                                                                }
                                                            }
                                                        }
                                                    }
                                                }
                                            }
                                        }
                                    }
                                }
                            }
                            else
                            {
                                QString dataType = ifcValueType_to_dataType(ifcValueType);
                                if(!dataType.isEmpty())
                                {
                                    property.insert("typeifc", ifcValueType);
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

WTF!?

KEEP IN MIND THAT I'M
SELF-UGHT, SO MY CODE
MAY BE A LITTLE MESSY.

LEMME SEE-
I'M SURE
IT'S FINE.



...WOW.

THIS IS LIKE BEING IN
A HOUSE BUILT BY A
CHILD USING NOTHING
BUT A HATCHET AND A
PICTURE OF A HOUSE.



IT'S LIKE A SALAD RECIPE
WRITTEN BY A CORPORATE
LAWYER USING A PHONE
AUTOCORRECT THAT ONLY
KNEW EXCEL FORMULAS.



IT'S LIKE SOMEONE TOOK A
TRANSCRIPT OF A COUPLE
ARGUING AT IKEA AND MADE
RANDOM EDITS UNTIL IT
COMPILED WITHOUT ERRORS.

OKAY, I'LL READ
A STYLE GUIDE.

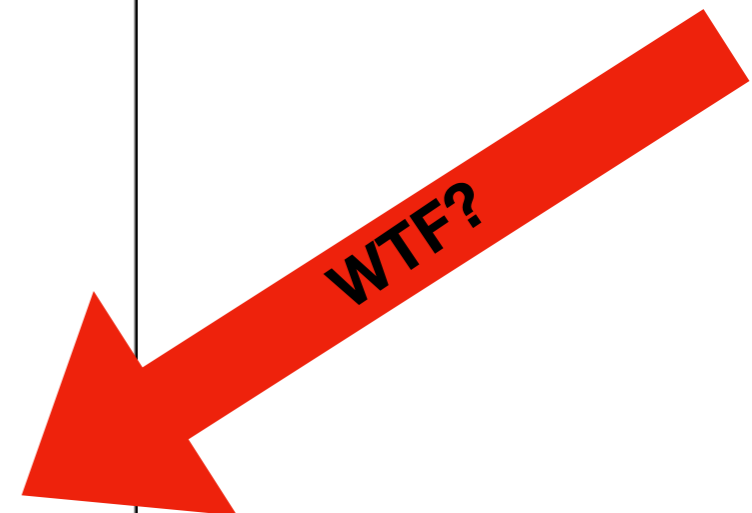
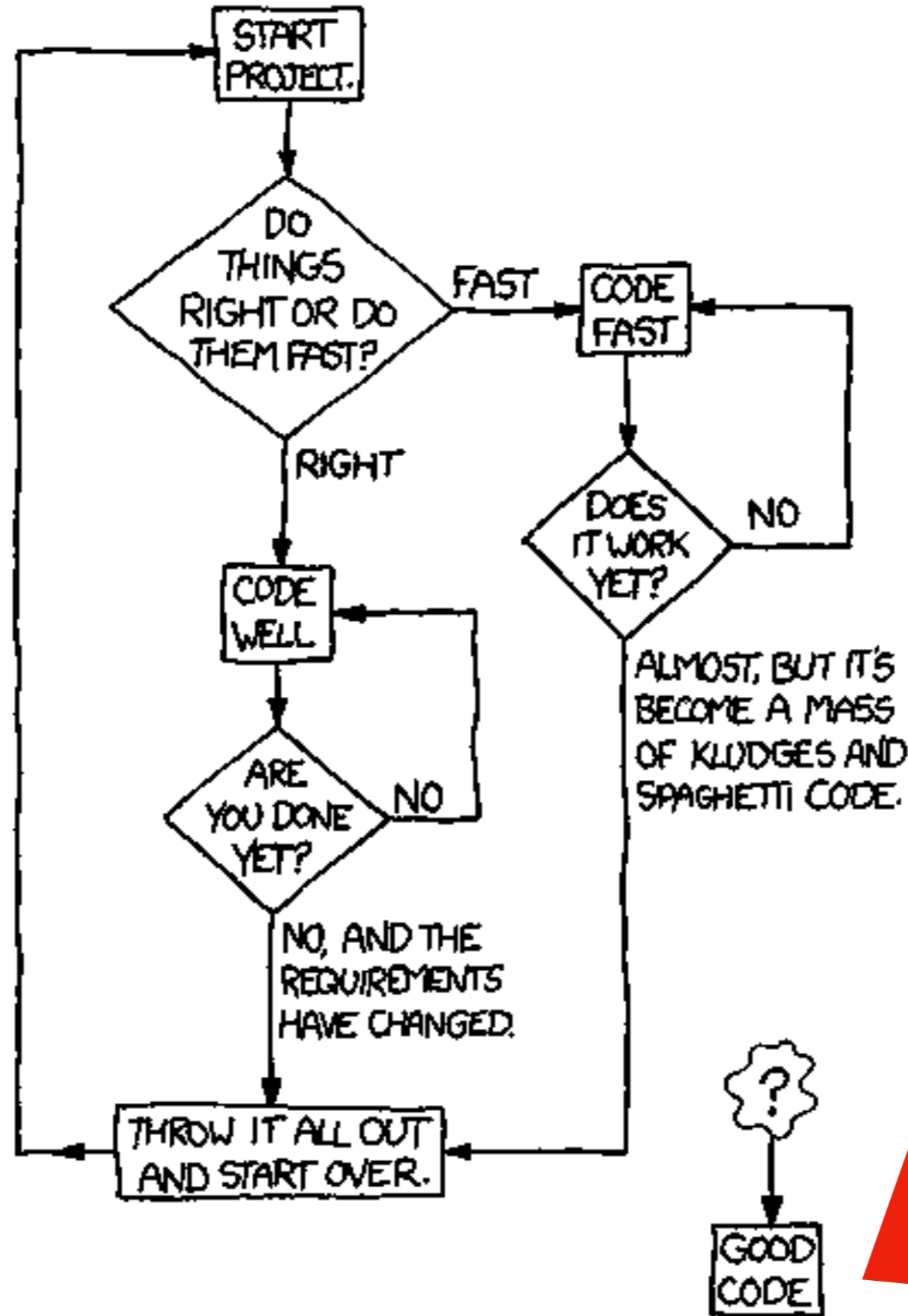


```

function canDoDungeon(profile, type, master, level){
  return (("14" in profile.dungeons.dungeon_types[type]
.tier_completions ? 15 : ("13" in profile.dungeons.dungeon_types
[type].tier_completions ? 14 : ("12" in
profile.dungeons.dungeon_types[type].tier_completions ? 13 :
("11" in profile.dungeons.dungeon_types[type].tier_completions ?
12 : ("10" in profile.dungeons.dungeon_types[type]
.tier_completions ? 11 : ("9" in profile.dungeons.dungeon_types
[type].tier_completions ? 10 : ("8" in
profile.dungeons.dungeon_types[type].tier_completions ? 9 : ("7"
in profile.dungeons.dungeon_types[type].tier_completions ? 8 :
("6" in profile.dungeons.dungeon_types[type].tier_completions ?
7 : ("5" in profile.dungeons.dungeon_types[type]
.tier_completions ? 6 : ("4" in profile.dungeons.dungeon_types
[type].tier_completions ? 5 : ("3" in
profile.dungeons.dungeon_types[type].tier_completions ? 4 : ("2"
in profile.dungeons.dungeon_types[type].tier_completions ? 3 :
("1" in profile.dungeons.dungeon_types[type].tier_completions ?
2 : ("0" in profile.dungeons.dungeon_types[type]
.tier_completions ? 1 : 0)))))))))) > level + (master ? 7 :
0)) && ((skillxplvls, skillxp) => {for (var i = 0; i <
skillxplvls.length; i++){if (skillxp < skillxplvls[i]){return
i-1;}};return skillxplvls.length-1;})(xplvls,
profile.experience_skill_combat) >= 15 && (((skillxplvls,
skillxp) => {for (var i = 0; i < skillxplvls.length; i++){if
(skillxp < skillxplvls[i]){return i-1;}};return
skillxplvls.length-1;})(dungeonxplvls,
profile.dungeons.dungeon_types.catacombs.experience) >
dungeonreqs[type][level + (master ? 7 : 0)]);
}

```

HOW TO WRITE GOOD CODE:



WTF!?

SOLID++: Least Astonishment (a.k.a. WTF)

People are part of the system. The design should match the user's experience, expectations, and mental models.

We are software developers, so the principle says that we shouldn't mislead the developers who will work on the code in the future with namings that are not coherent with the behavior inside the constructs.



Lesson 4

TRAINING PROGRAMME

SOLID++: Least Astonishment (a.k.a. WTF)

```
int Multiply(int a, int b)
{
    return a + b;
}

int SaveFile(string filename, string text)
{
    Console.WriteLine(text);
}

public class SomethingFactory {
    public void DoSomething()
    {
        //write on Database
    }
}
```



Lesson 4

TRAINING PROGRAMME

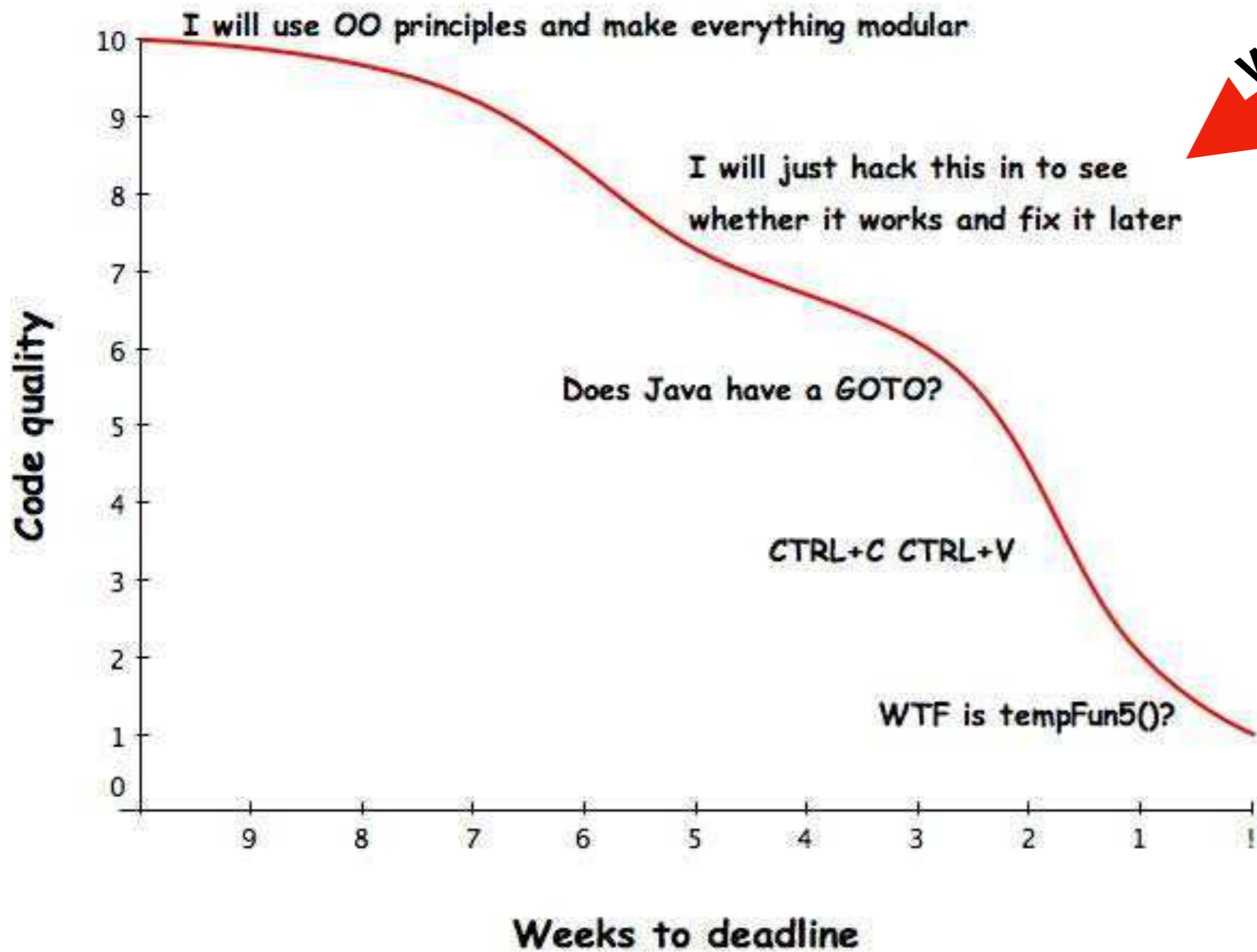
ALCOR
academy



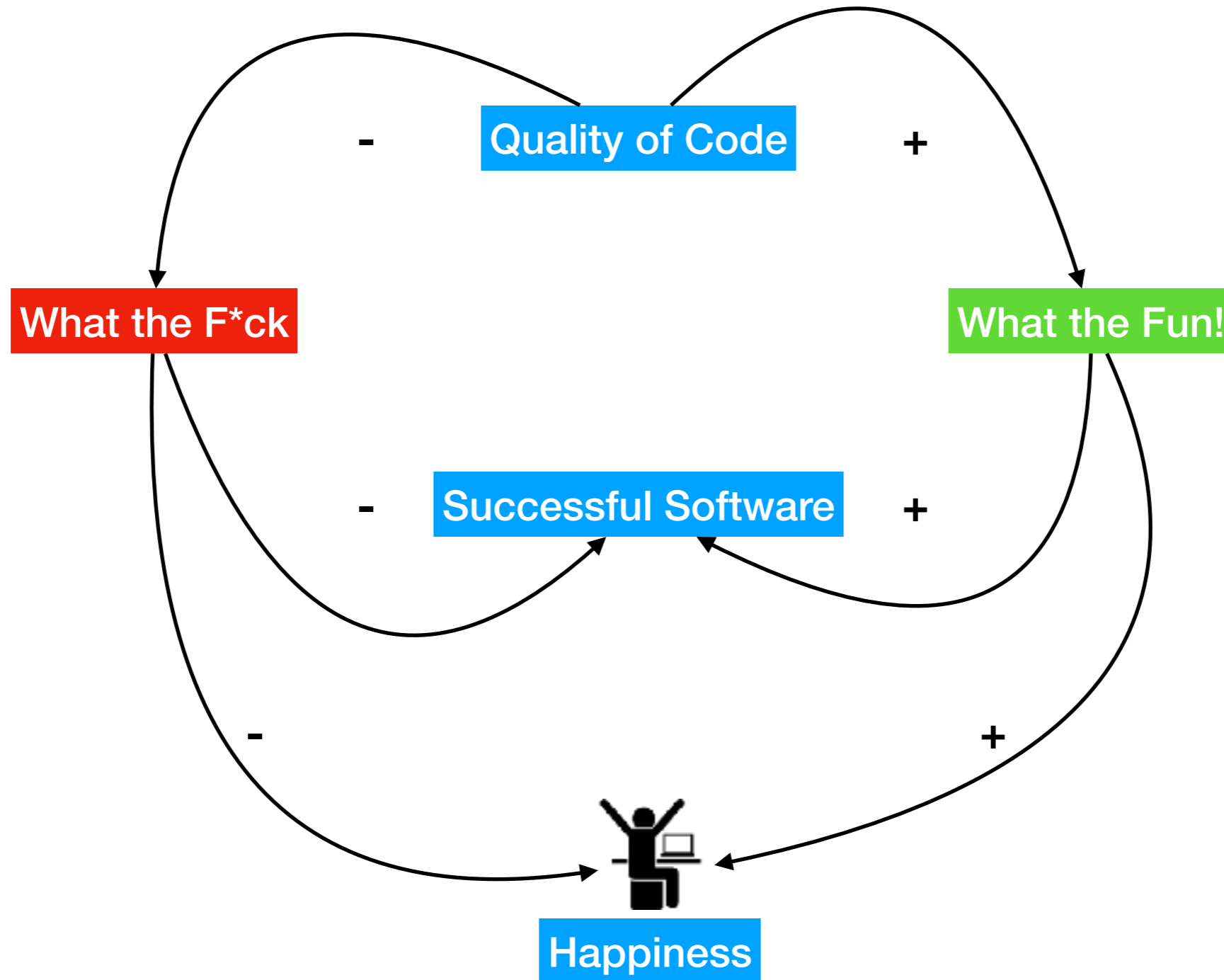
majcon



WTF!?



From WTF to WTF!



**What's special about Superman is
that he will always make the
right choice.**

 SayingImages.com

Max Landis



MARCO CONSOLARO



ALESSANDRO DI GIOIA

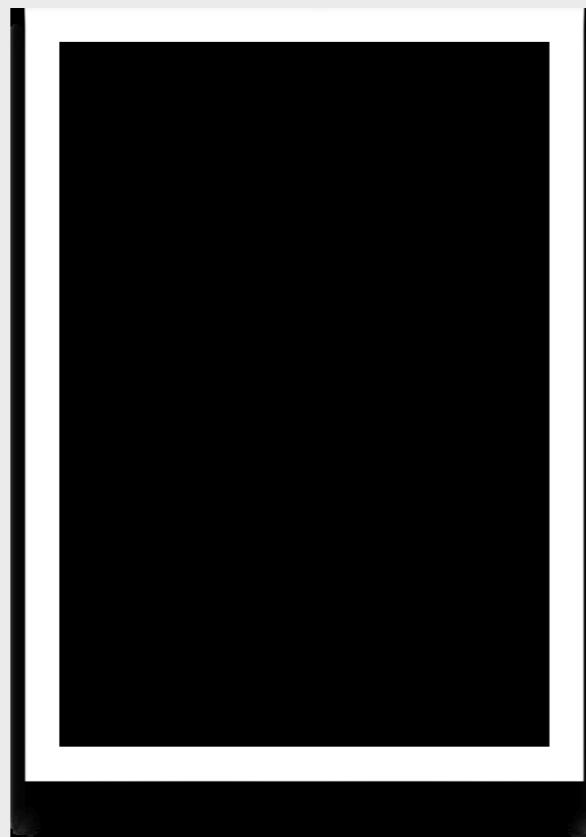
Thanks for attention!



All Credits for the good stuff to



**All other issues..
> contact me directly**



 Majer Consulting

Damir Majer

www.majcon.de

Twitter: @majcon