



# Evendo

Stay organized.

# Agenda

---



INTRODUCTION



WHY SHOULD  
YOU TAKE US?



PROJECT  
MANAGEMENT



TECHNOLOGIES



ARCHITECTURE



QUALITY  
ASSURANCE



DEMO



FUTURE  
PLANNINGS

# What happens if



9 MONTH'S OF  
DEVELOPMENT



3 MOTIVATED  
DEVELOPERS



WITH 1 PASSIONATE  
PROJECT MANGER



A GREAT IDEA

comes together?

# Evendo



# Advantages of Evendo



STAY ORGANIZED



NEVER FORGET  
ATTENDING TO A  
MEETING



SYNCHRONIZE YOUR  
CALENDAR WITH OTHERS



ADD TODO'S TO  
APPOINTMENTS

# Why should you take us?

Image a team of 3 person **without** Evendo:



Your team takes a lot of time to plan their meetings.



But maybe your colleague missed to prepare for the meeting because of stress or missed to finish something before



The productivity of your company drops

# Why should you take us?

Image a team of 3 person **with** Evendo:



Your Team plans their meeting  
easily with evendo



Your colleague added some  
TODO's to this appointment  
before. Everybody is prepared  
now.



The meeting was great! The  
productivity of your company  
increases!

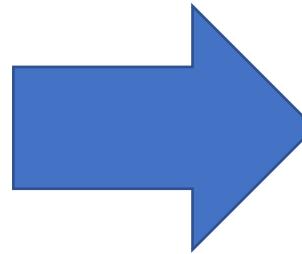
# But what's that in numbers?



Your team:  
3 developers with an  
hourly rate of 45€



Our Internal studies  
resulted an productivity  
increasement up 25% by  
using Evendo



Your price for Evendo Software:  
1999€ + 150€ monthly fare.

If you take an advantage of 25%  
of your employees productivity  
measured on their hourly rate:

Evendo will be payed off after  
**only 1 month**

# Project Management

---



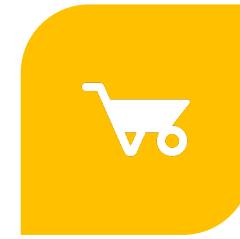
# SCRUM – Why do we use it?



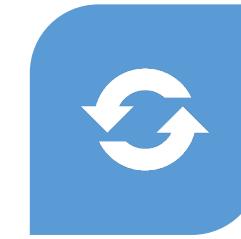
SAVE TIME AND  
MONEY



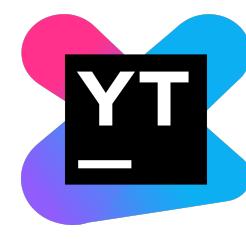
ENCOURAGES  
TEAMWORK



EASY TO WORK  
WITH



AGILE TO MAKE  
CHANGES



TOOL:  
YOUTRACK

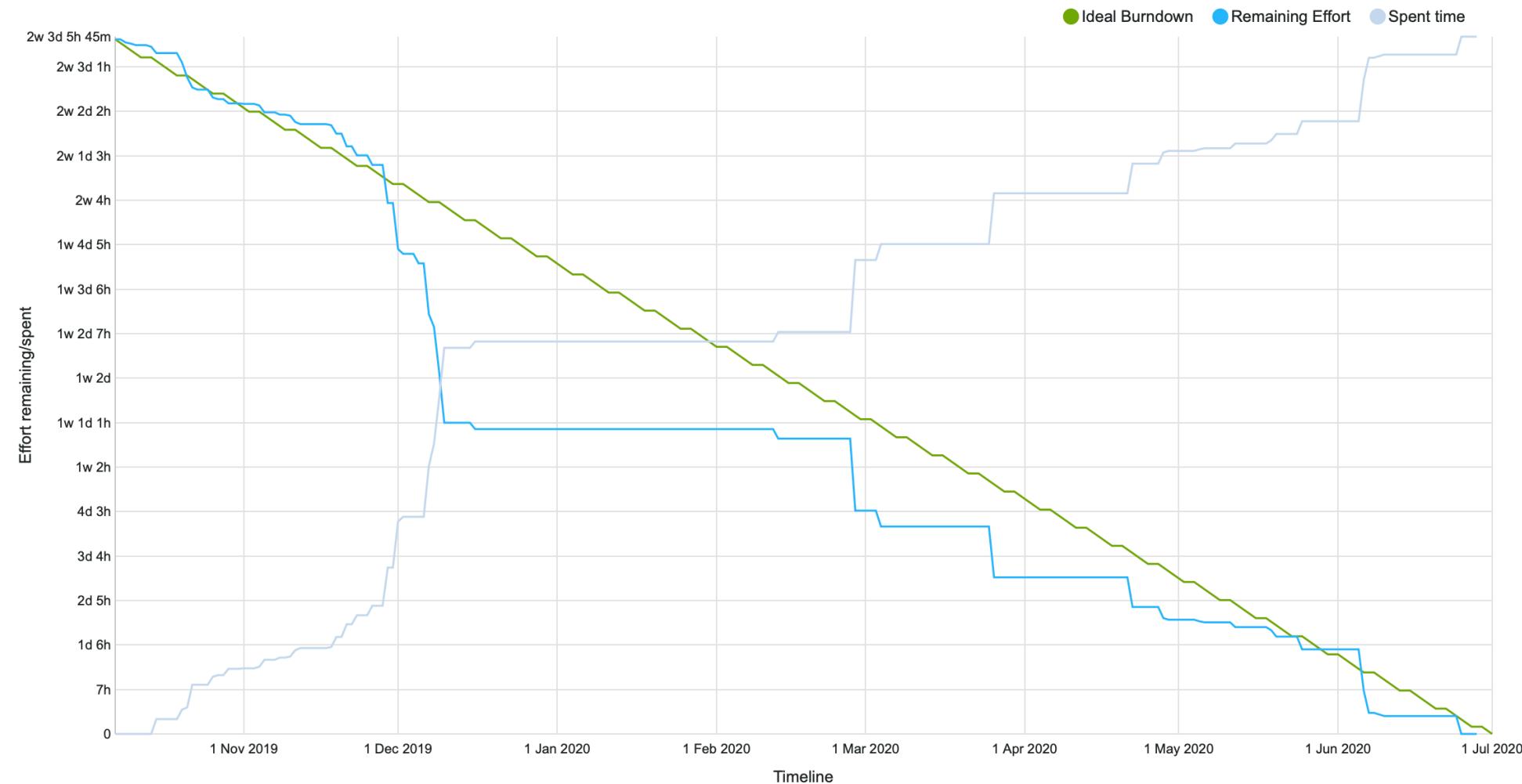
# Our YouTrack

The screenshot shows a backlog of cards in the YouTrack application. The cards are categorized under 'Uncategorized Cards'. Each card includes a title, description, assignee, priority, type, and subsystem. There are also time tracking and status indicators.

Card ID	Title	Description	Assignee	Priority	Type	Subsystem	Time Tracking	Status
EVENDO-61	OnPressAction for Events	Events should display all assigned todos onClick	KG	Normal	Task	No Subsystem	5h	In Progress
EVENDO-112	[Backend] Code coverage	We needed to add the code coverage, fixes some issues (by Ms. Berkling)	NS	Normal	Task	No Subsystem	2d	Fixed
EVENDO-62	Add CreateTodo Activity	Todos should be created similarly to events (or should they? @nschmuck @Marius)	KG	Normal	Task	No Subsystem	6h	Verified
EVENDO-113	Add RUP Workflow to old issues		MA	Normal	Task	No Subsystem	10m	Open
EVENDO-109	Chenge folderstructure for UML and UseCases		MA	Normal	Task	No Subsystem	30m	In Progress
EVENDO-16	Design Thinking General X	create a design prototype for the UIs of the app	NS	KG	+1	Minimal Task No Subsystem	2h	Fixed
EVENDO-110	Write Risk-Management Document		MA	Normal	Task	No Subsystem	1h	Verified
EVENDO-111	Write Function Points Document		MA	Normal	Task	No Subsystem	1h	Open
EVENDO-108	Write jUnit-tests		MA	Normal	Task	No Subsystem	3h	In Progress

# YouTrack Burndown Chart

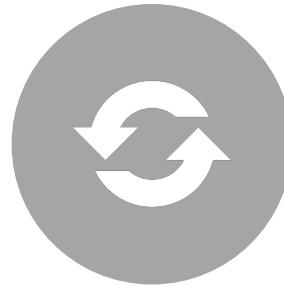
Burndown



# RUP – What is that?



Rational Unified Process



Iterative Development  
Process

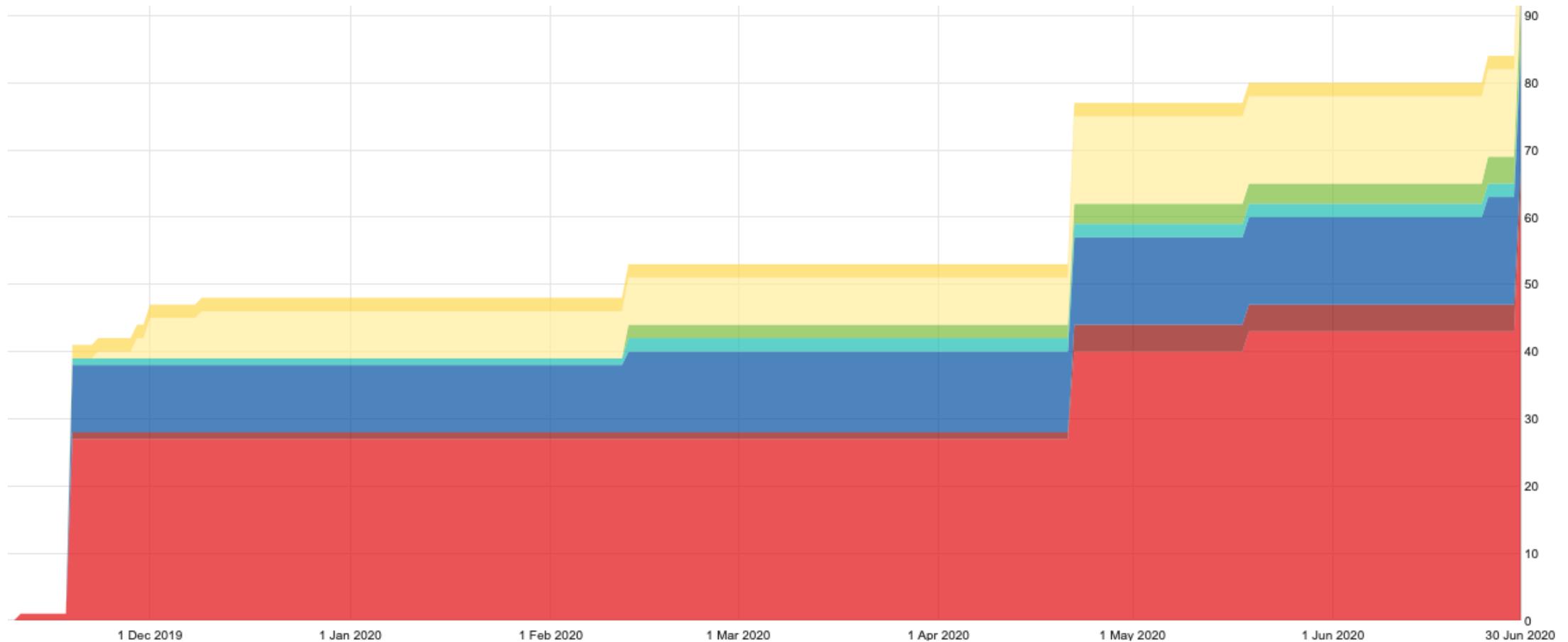


Categorize Tasks into  
groups



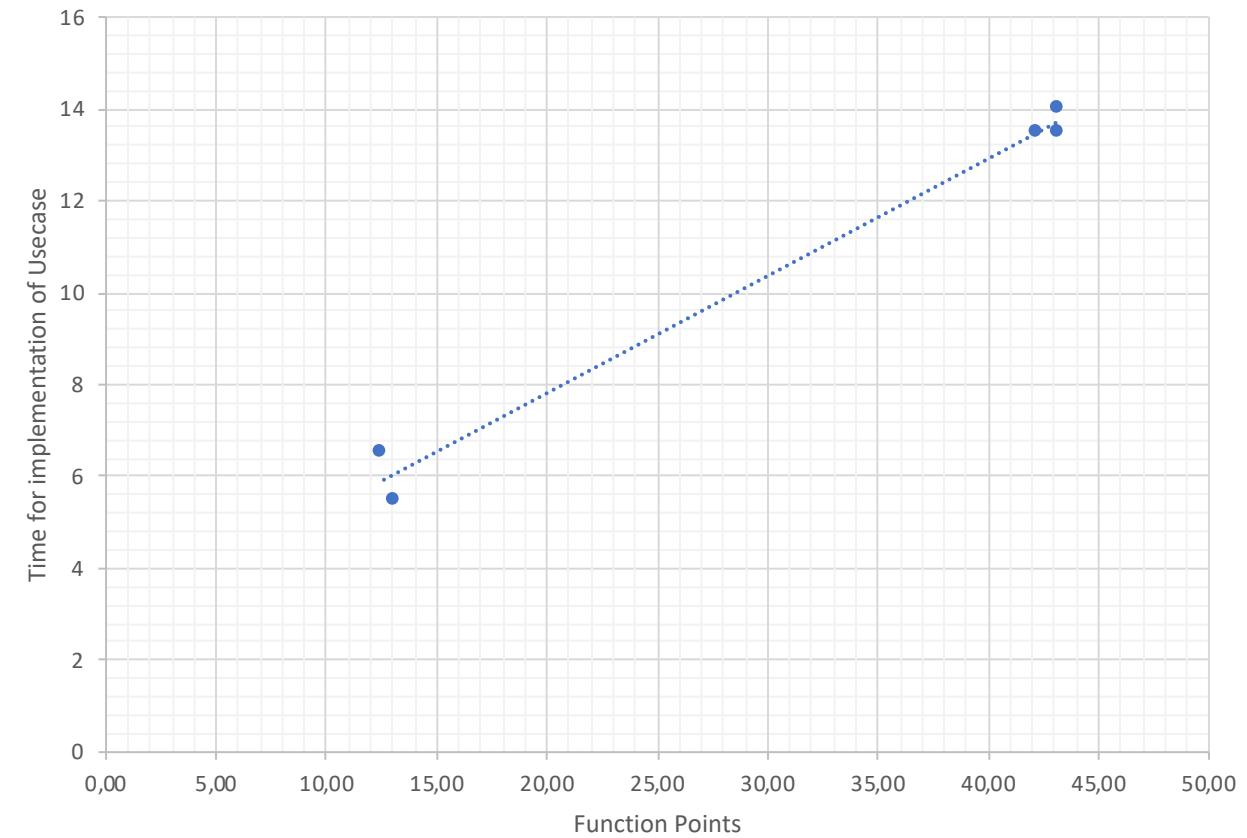
Continuous Quality  
Assurance

# Our RUP Chart



# Time and Function Points - Costs

Users	Time estimated	Time spent
<b>Total time</b>	<b>175h 30m</b>	<b>109h 45m</b>
KG - Kolja Groß	21h 20m	
MA - Marius	37h 39m	
NS - nschmuck	50h 46m	



# Costs in detail

A new usecase with 30 FP will approximately take 11 hours.

Our company:

Hourly rate of us: 18€

Estimated price: 198€

Other companies\*:

Hourly rate for a software engineer: ~ 45€

Estimated price: 495€

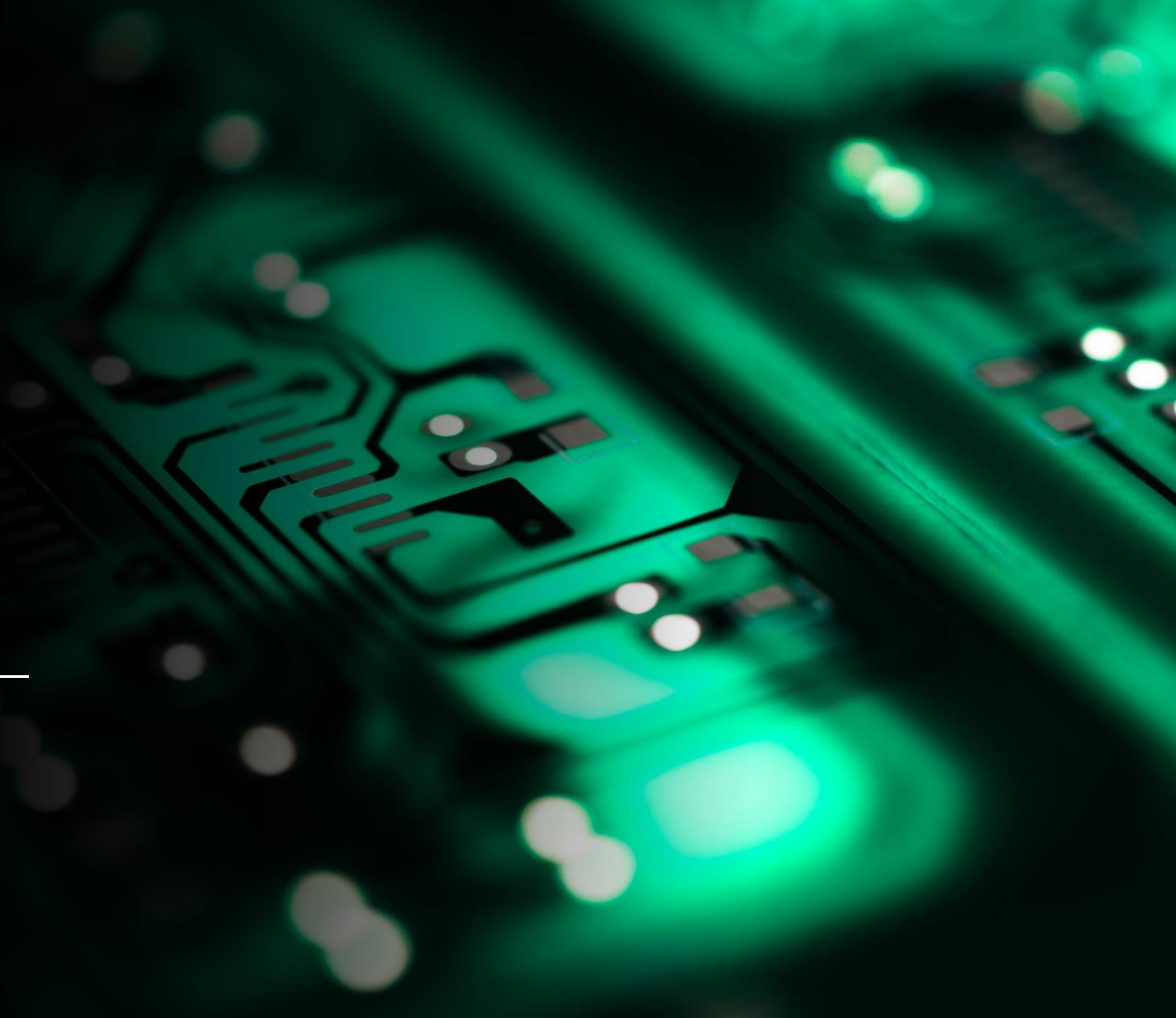
You will save 297€

\* Estimation on gulp for a Software Engineer



# Technologies

---



# Technology overview

---

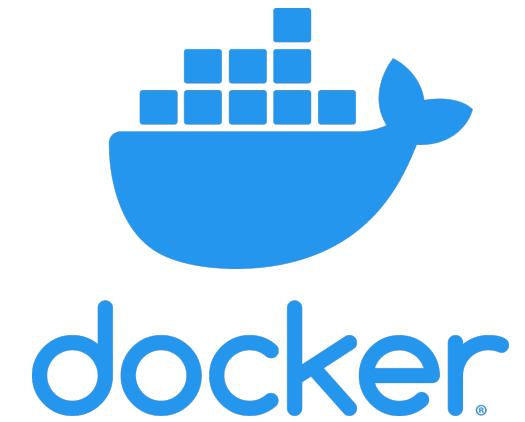
# Frontend Technology Overview

---



# Backend Technology Overview

---



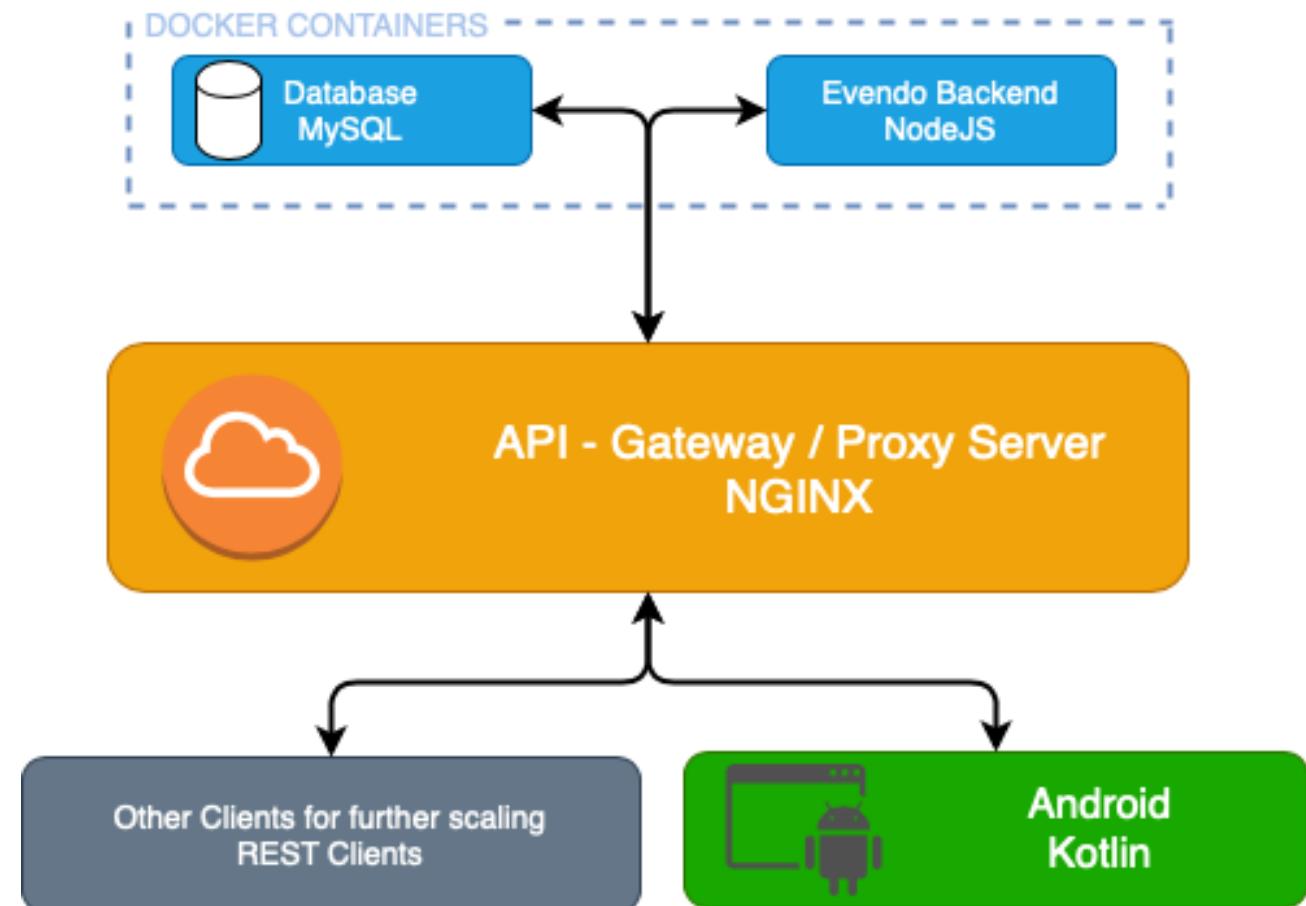
# Technology Stack Diagram

We are using a three-tier architecture

Data & Backend Server Layer

Logic Layer

Presentation Layer



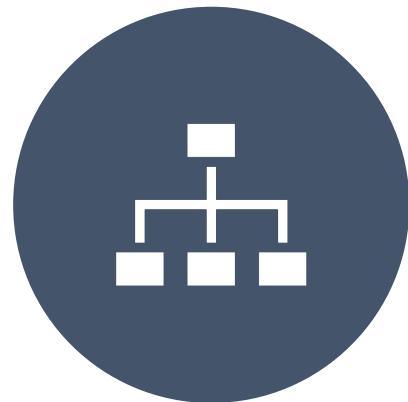
# But what's the profit?



LOWER COMPLEXITY INSIDE THE  
SYSTEM



SCALABILITY BECAUSE OF OUR  
PROXY SERVER LAYER AND DOCKER



CLEARLY DEFINED STRUCTURE

# Quality Assurance

---



Quality is more important than quantity. One home run is much better than two doubles.

---

Steve Jobs

# Why is this so important?



For you it should be so easy  
to maintain our code



Performance is no wish, it is  
a requirement

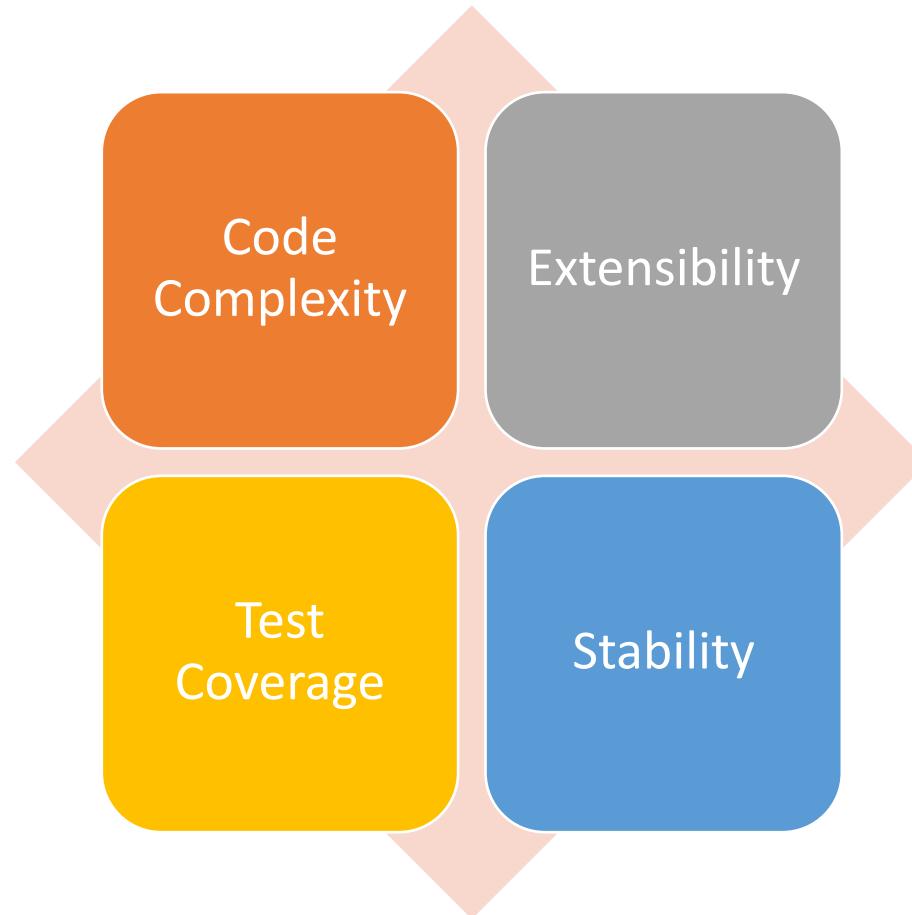


Time is money



Tool should work all time

# Our definition of Code Quality



# Code Complexity

## Frontend

## Backend

Complexity Index: Lower scores are better

1

Example Backend files their complexity Index

5

2

2



Explanation: The Grade is from A – F. A is the best, F the worst.



# Insertion: Testing



# Testing Frontend



Cucumber tests for incorrect user inserts and internal convert of Variables

JUnit tests functions for successful execute

Espresso tests our UI to keep it working

**Continuous Integration:**  
Runs every time while compiling the Frontend Application (APK)



## Testing Backend

---

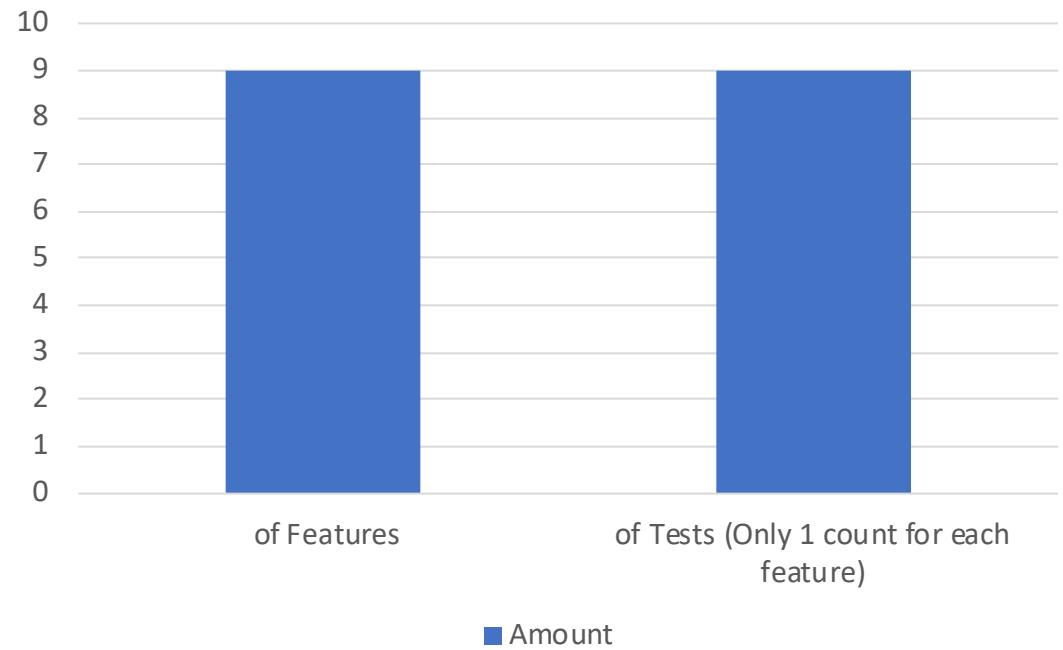
- Jest
  - Framework for automated Testing
  - Big companies like AirBnB swear on that
- Automation: runs every time on backend restart

# Test Coverage

**Frontend**

**Backend**

Amount



# Test Coverage

**Frontend**

**???.?%\***

**Backend**

**97.67%\***

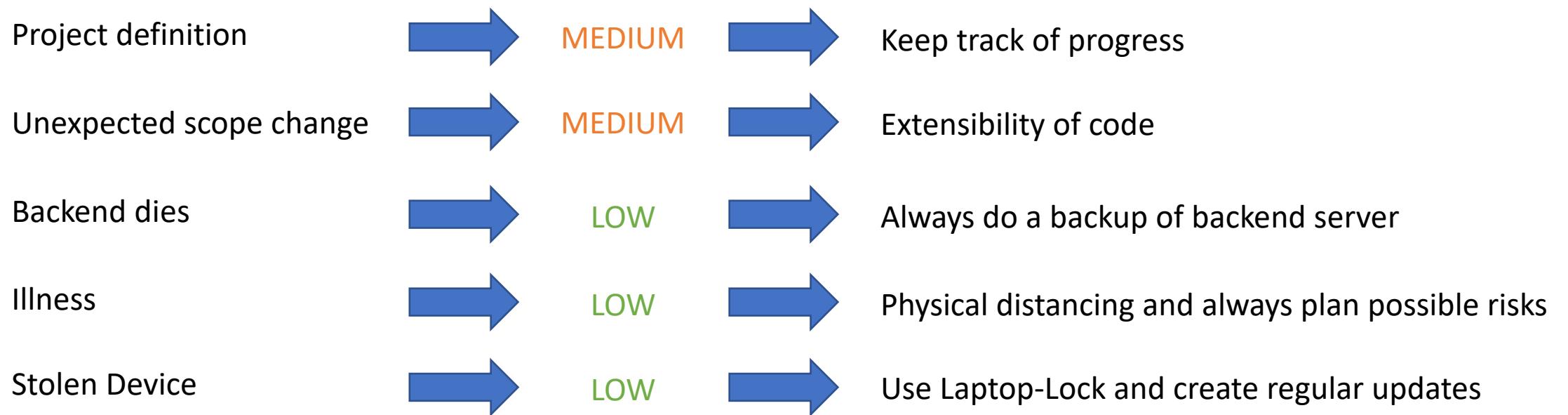
\*Relative Result is measured in percentage of full functions

# Stability

- We performed different stability tests over the last months
  - "Stress Tests"
  - Tests on idle (when nobody is using server performance)
  - "Crash Tests"
- And our result:
  - Evendo's Backend application is running for nearly three month's now!

16445.evendo (04/08/2020 01:27:24 PM)

# Risk Management





# Demo



# Future Plannings

---



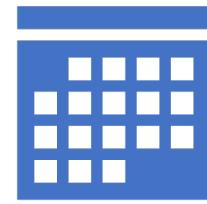
# Possible Usecases



Update Events



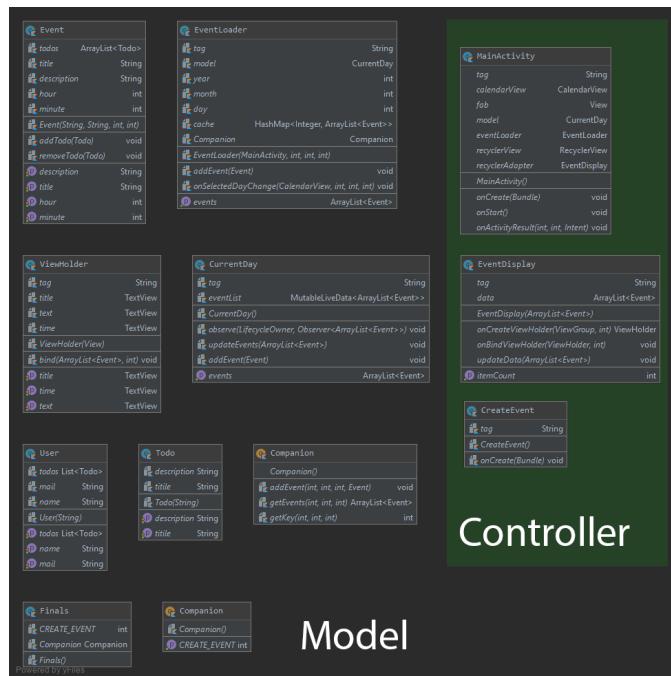
Update Todos



Import Calendars

# Class Diagram

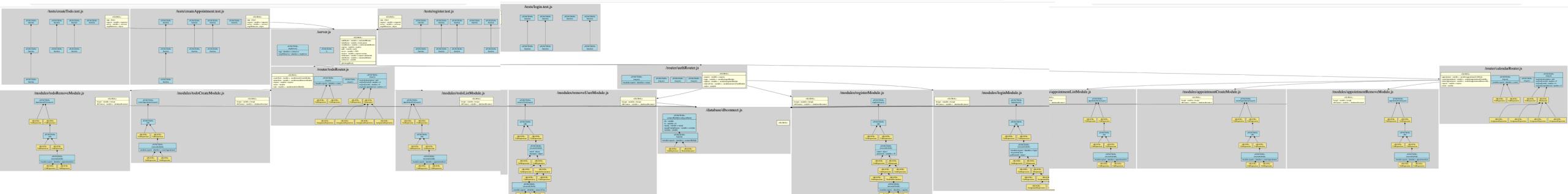
Frontend Class Diagram:



Controller

Model

Backend Class Diagram:



# Observer Patterns

## Frontend:

Observer Patterns in type of Observable Objects are used in our Frontend App

## Backend:

Our Backend uses Observer Patterns for a great extensibility of new functions

## Advantages:

- Low Costs
- Low Complexity

Thanks for  
listening

---

