

Mathinator

1 / 20

An approach to character recognition and solving math expressions

Teammitglieder

2/20

- Sascha Hug
 - Projektleiter
 - Entwickler
- Tobias Lamm
 - Entwickler
 - Softwarearchitekt
- Tim Saupp
 - UC Designer
 - Testmanager
 - Entwickler

Agenda

3 / 20

1. Vision
2. Umsetzung
3. Demo

Unsere gemeinsame Vision

4/20

Intuitiv. Smart. Kundenorientiert.

Klassischer Taschenrechner.

Handschrifterkennung.

Verlaufsfunction.

Use Cases

6/20

Handschrifterkennung

Vom Bild

Vom
Kamera-
Livefeed

Bild bestätigen

Der Nutzer kann bestätigen, ob das aufgenommene Bild benutzt werden soll

Verlaufsfunction

Zeige Listen-Menü mit bisherigen Einträgen

User kann Einträge löschen

Manueller Taschenrechner

Der User kann manuell Gleichungen eingeben, die vom Taschenrechner gelöst werden

Hilfefunktion

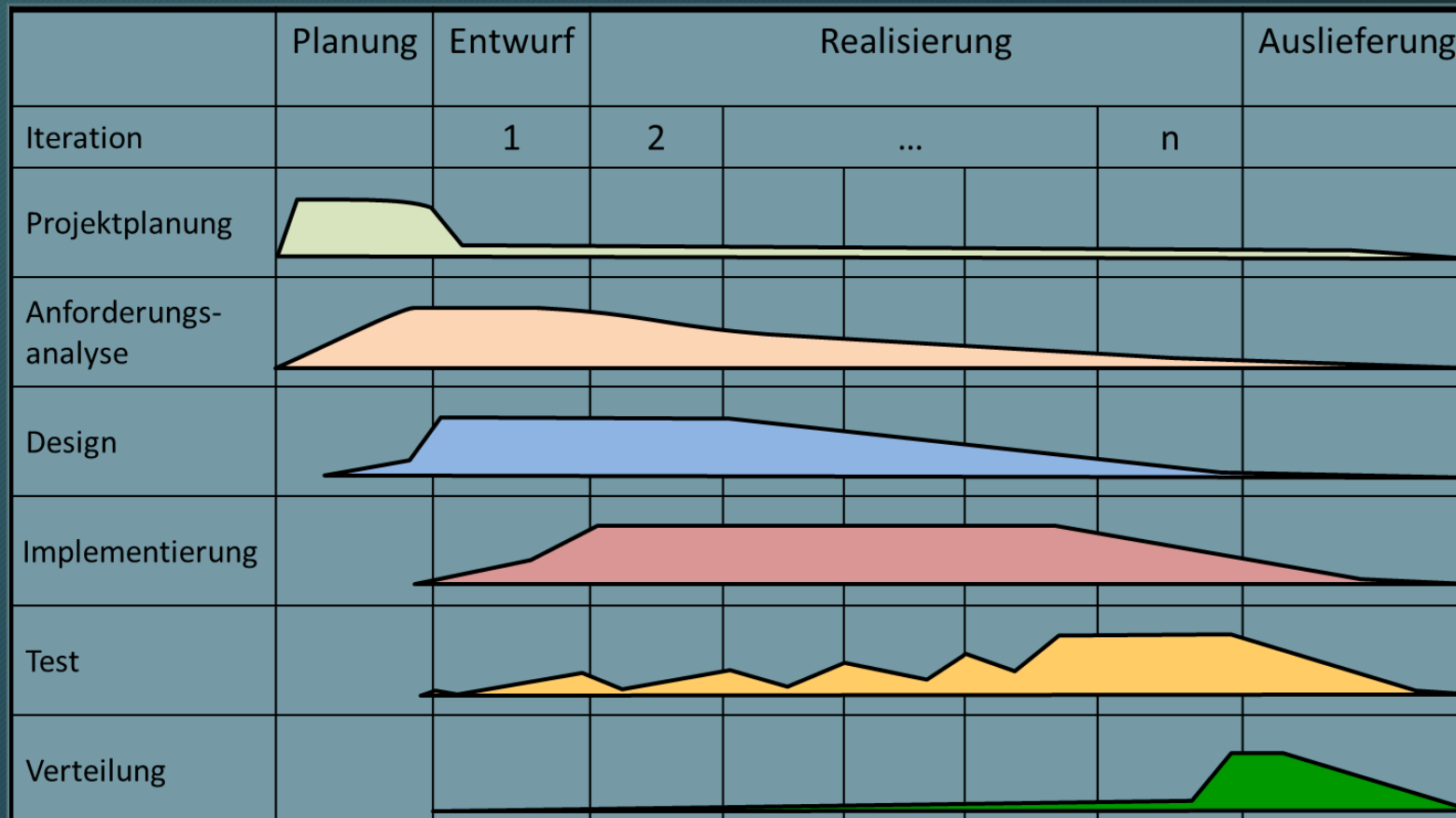
Beim ersten Start der App wird der Nutzer durch die App geführt

Umsetzung

7/20

RUP und SCRUM im Dialog

8/20



Softwareentwicklungsprozess

9/20

Scrum

- Flexibilität.
- Transparenz.
- Zeitnahe Umsetzung gesetzter Ziele.

My Dashboard ▾

1 of 2 → Dashboard

issues sort by: updated 54

N

MATHINATOR-54 Control

N

MATHINATOR-53 planning the next Steps

N

MATHINATOR-1 User angelegt

N

MATHINATOR-52 Preparations for mid Term Presentation

N

MATHINATOR-42 Generate ANTLR Lexer, Parser, BaseVisitor from grammar file

N

MATHINATOR-51 [Add general description for each class](#)

N

MATHINATOR-50 Reload ListView after deleting entries

N

MATHINATOR-49 Update CRUD operations to allow deletion of single entries

N

MATHINATOR-48 Implement ActionMenu to allow User to delete entries

N

MATHINATOR-47 Pass necessary Information from HistoryActivty to HistoryEntryActivity to View selected Entry in Submenu

N

MATHINATOR-46 Update Database Scheme to allow persisting the Result of an equation to the dataset

N

MATHINATOR-45 Create Submenu for Viewing selected entries

N

MATHINATOR-25 Write Comments for ListView Adapter

N

MATHINATOR-24 Test CRUD Operations of SQLite DB

N

MATHINATOR-44 Feed Input String from App to parser and retrieve / show the result

N

MATHINATOR-43 Create BaseVisitorImplementation Class for Parser

N

MATHINATOR-41 Define first draft of the grammar for arithmetic expressions

N

MATHINATOR-40 Implement ANTLR Parser Generator for Evaluation of Arithmetic Expressions

N

MATHINATOR-26 Persist Entries from the Manual Calculation Menu to SQLiteDB

N

MATHINATOR-39 Implement the Camera

N

MATHINATOR-38 Set Target/Compile API to 21

N

MATHINATOR-37 [Organise Android 7.0 Device](#)

N

MATHINATOR-27 [Create Robotium Testfiles](#)

N

MATHINATOR-28 [Testing App with Robotium](#)

N

MATHINATOR-36 [Use Case Diagram](#)

Offene Aufgaben

Total

6

Tim Saupp

350%

Unassigned

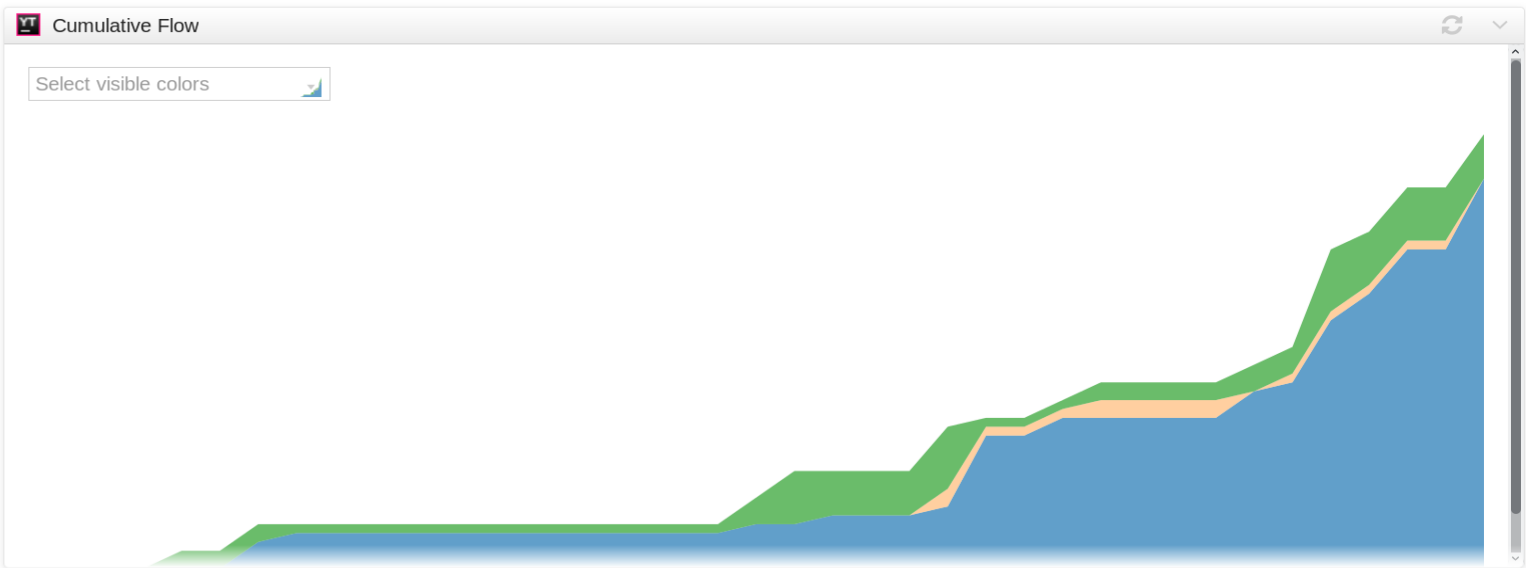
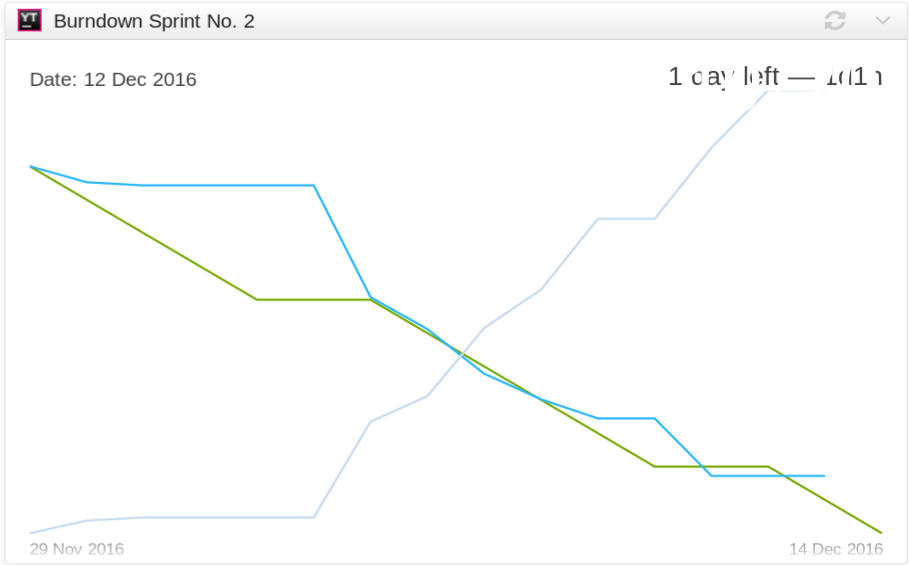
117%

Sascha Hug

117%

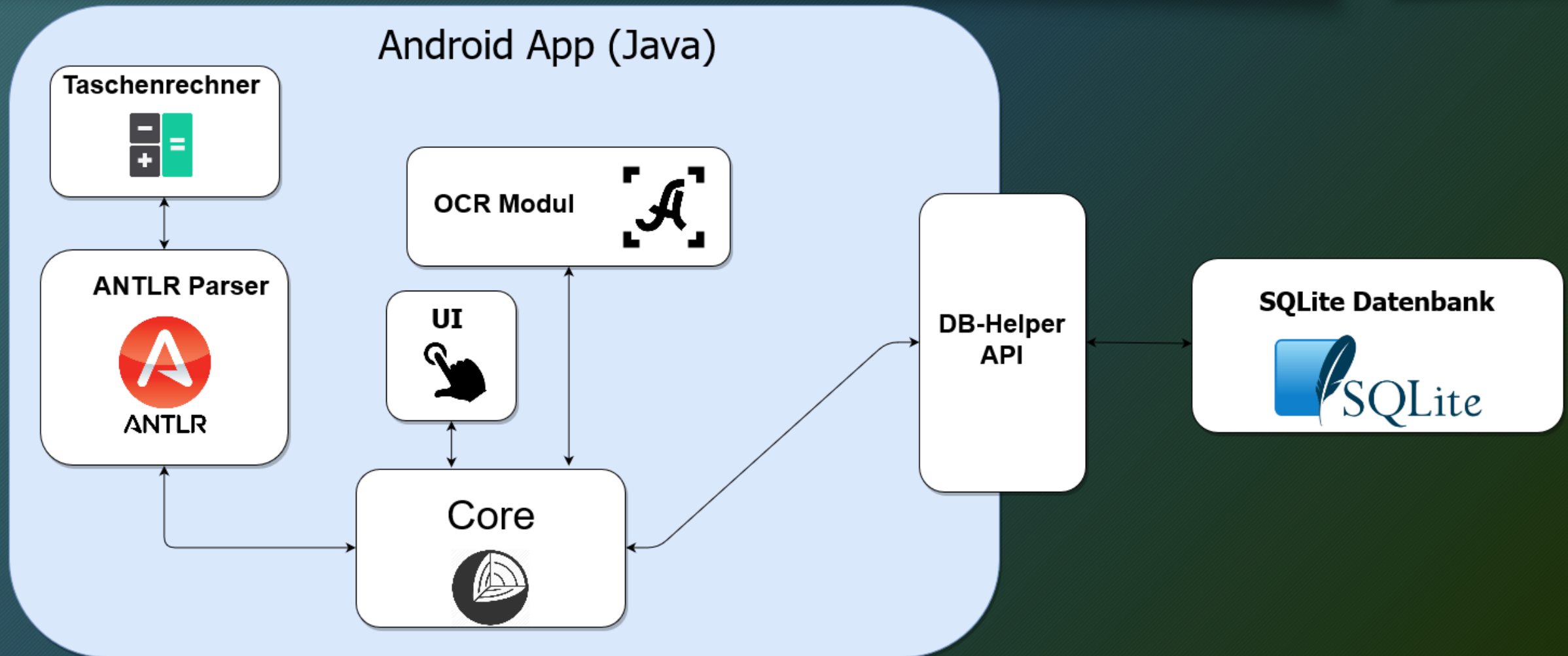
Tobias Lamm

117%



Technologien

11/20



Git

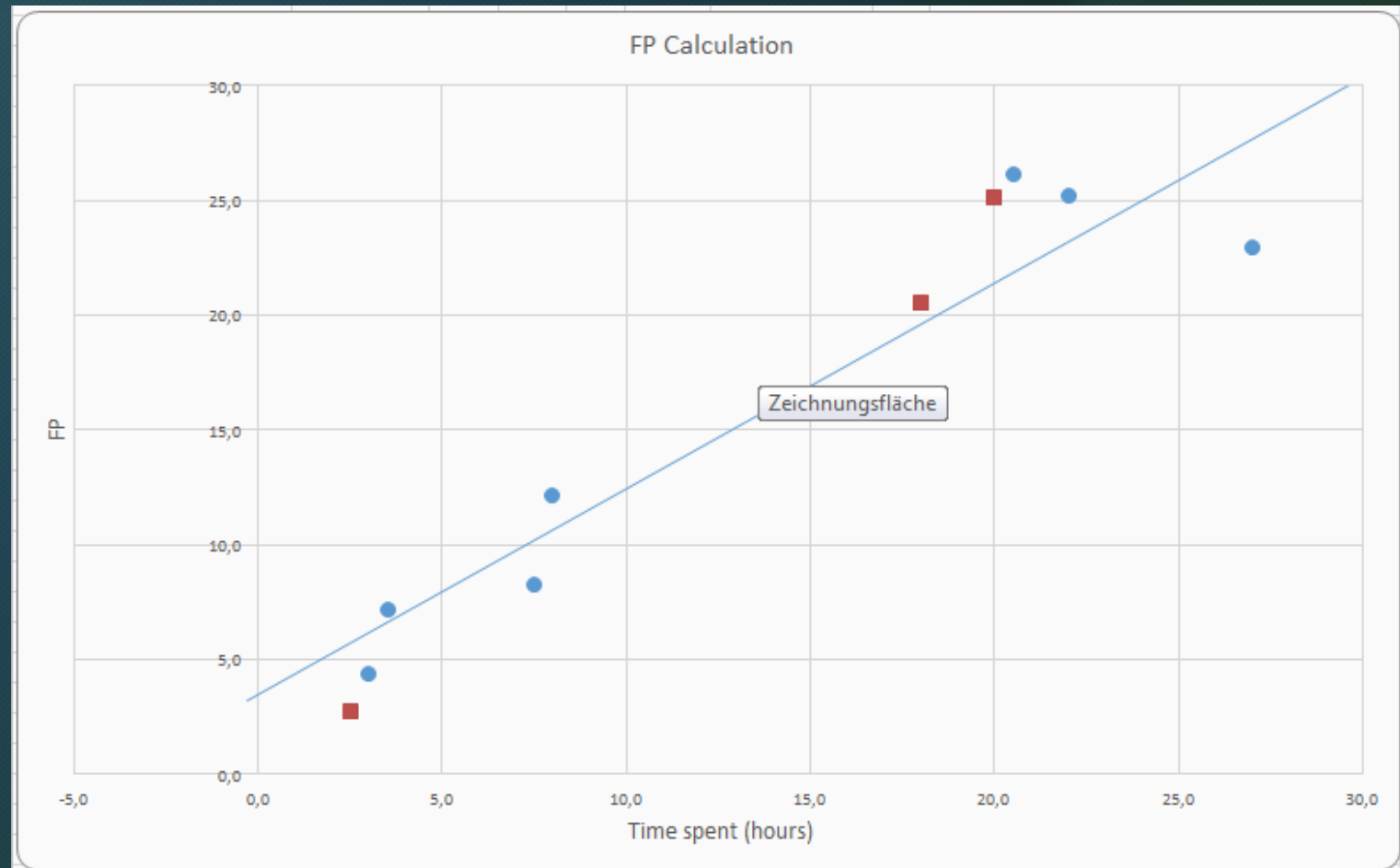
- Mehrere Branches. Lauffähige Version / Entwicklerbranch.
- Transparenz für Kunden.
- Optimierte Zusammenarbeit.

Langzeitplanung

13/20

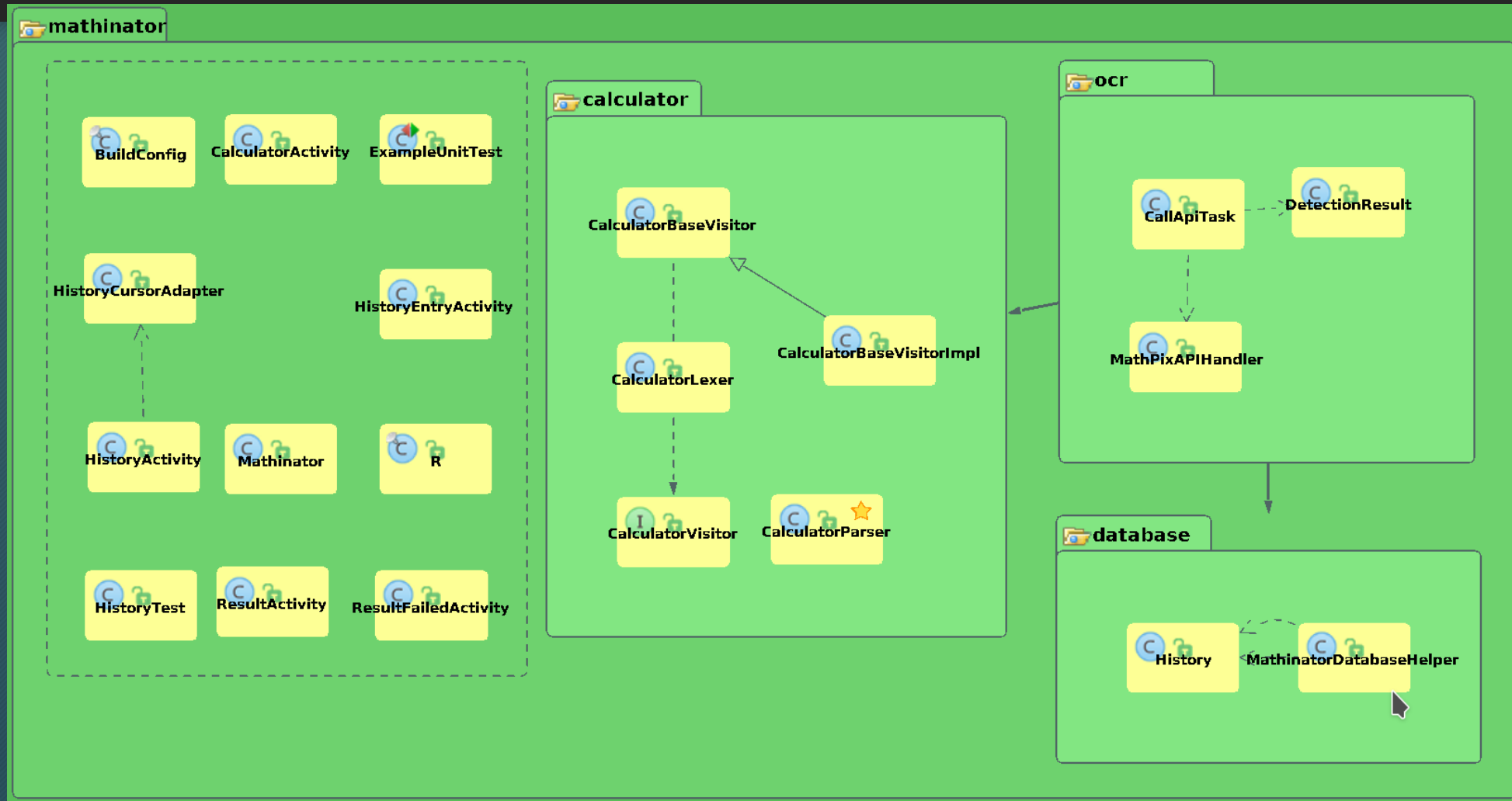
Mittels Function Points

- Kostenschätzung.
- Vergleich der Schätzung mit tatsächlicher Zeit.



Klassendiagramm

14/20



Metriken als Maßzahlen für Aspekte unseres Codes

- Komplexität.
- Kapselung.
- LCOM (Lack of Cohesion in Methods).

Interpreter Pattern

- Grammatik für arithmetische Ausdrücke.
- Problemlos erweiterbar.



Continuous Integration

17/20

SaschaHug / Mathinator 

build **passing**

[Current](#) [Branches](#) [Build History](#) [Pull Requests](#) > [Build #32](#) [Job #32.1](#)

✓ **master** Removed unused imports

 Commit 31507f6

 Compare 008f582..31507f6

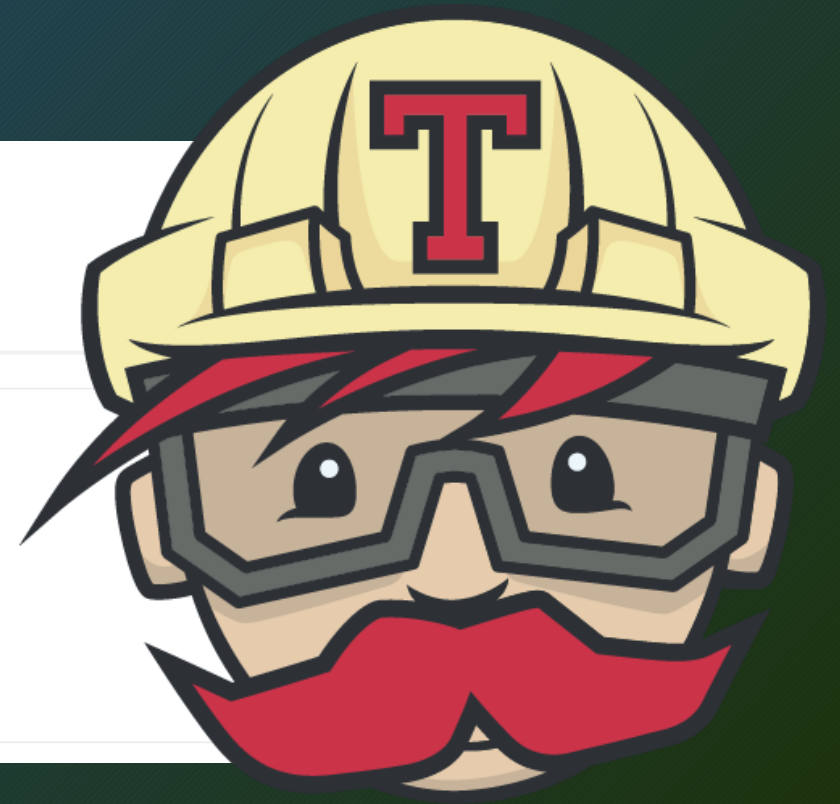
 Branch master

 IsThisTheFuture authored and committed

🔗 #32.1 passed

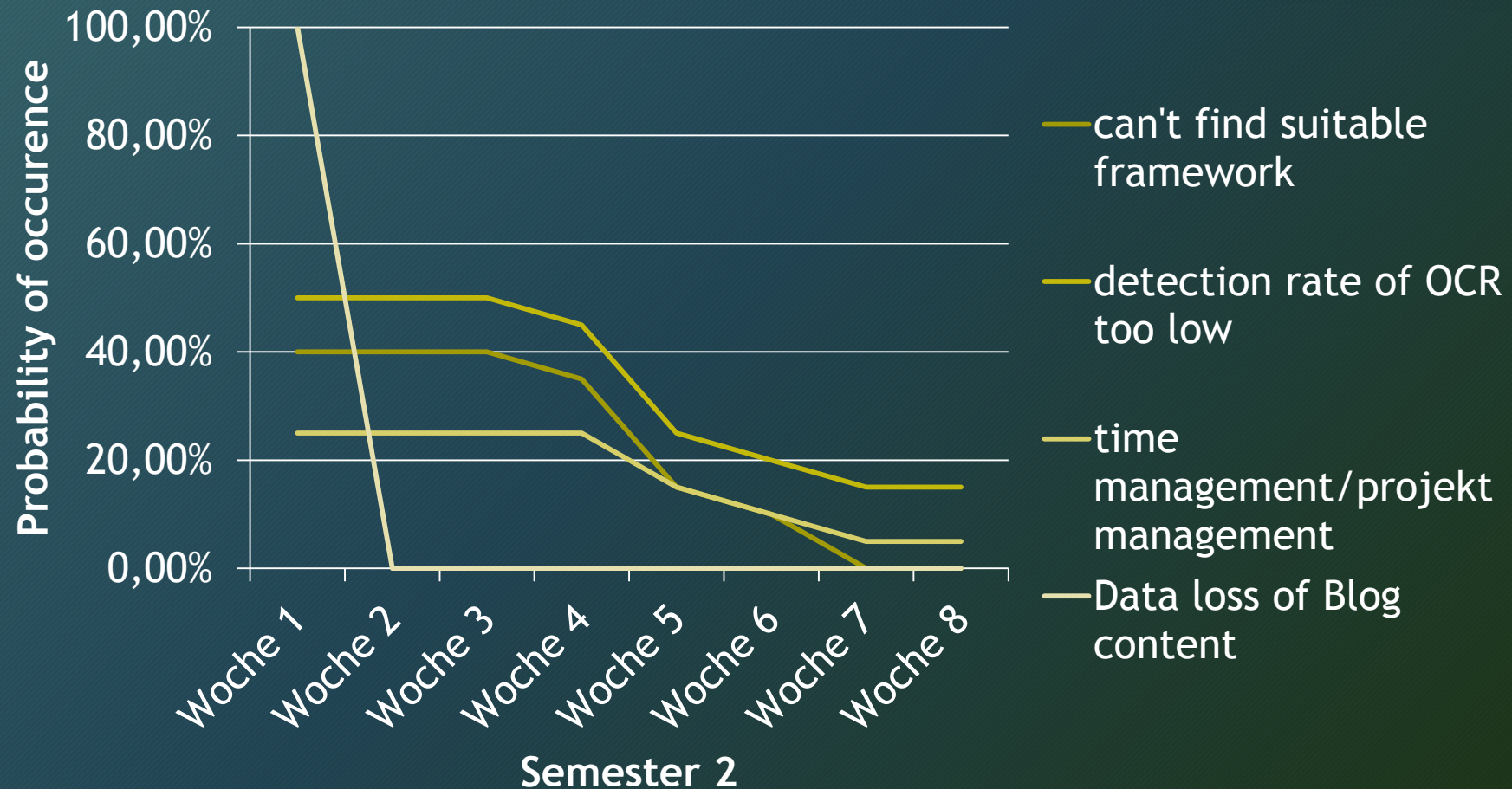
🕒 Ran for 21 min 27 sec

📅 about 5 hours ago



Riskmanagement

18/20



Demo

19/20

Danke für Ihre Aufmerksamkeit

20/20

Bei Fragen stehen wir Ihnen gerne zur Verfügung.