

# Campus-Routenplaner

## Praxis der Software-Entwicklung

Einführung · 6. Mai 2020

Guido Brückner · Michael Hamann · Paul Jungeblut

INSTITUT FÜR THEORETISCHE INFORMATIK · LEHRSTUHL ALGORITHMIK



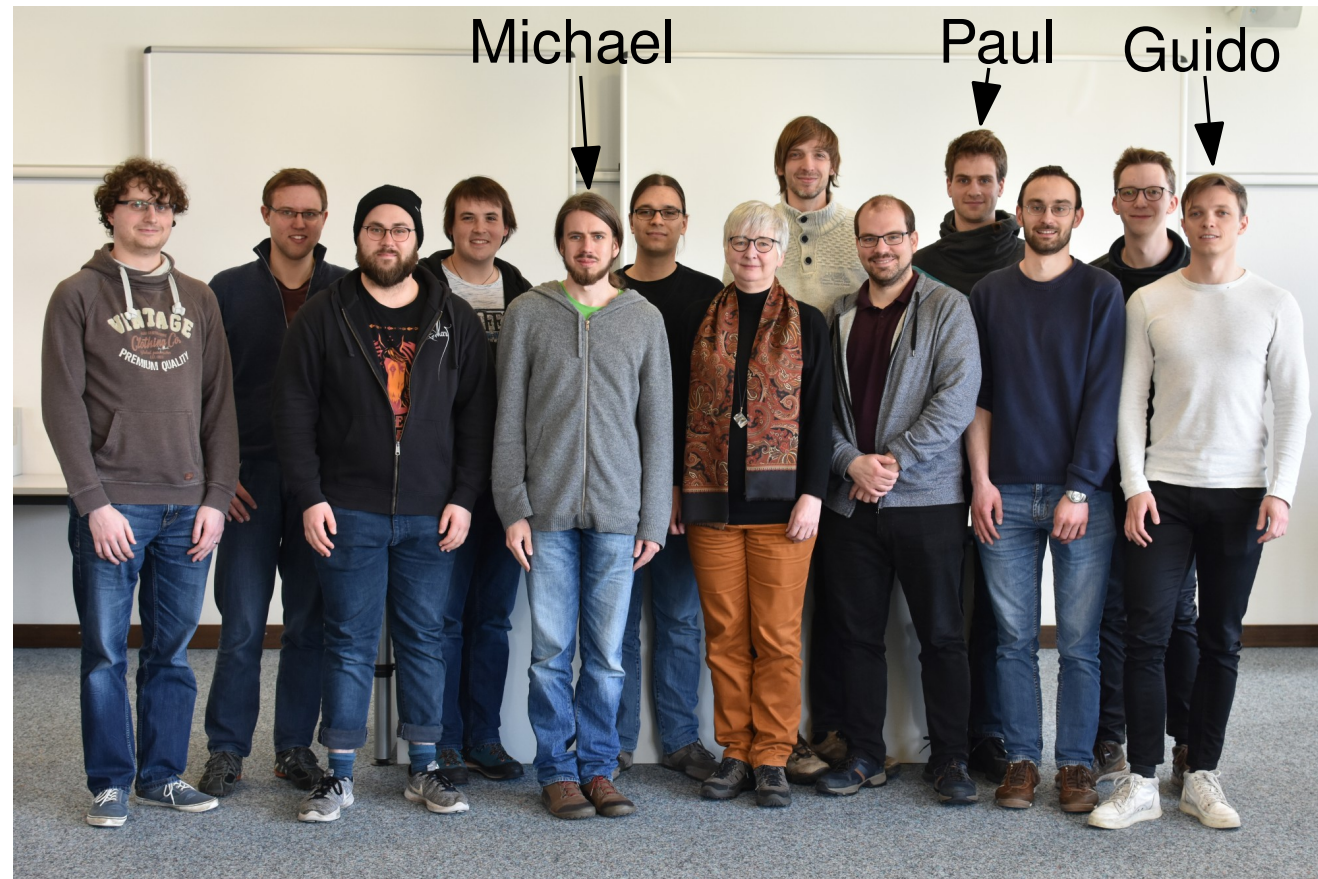
1. Organisation

2. Eure Aufgaben

3. Werkzeuge

# Vorstellung der Teilnehmer

Wir sind . . .



Wer seid Ihr?

- Name
- Wissen und Erfahrungen

# Zielsetzung

Programmieraufgaben

≈ 200 LOC

Grep

≈ 9.000 LOC

PSE – Campus-Routenplaner

≈ 10.000 LOC

MyPaint (Malprogramm)

≈ 52.000 LOC

Kate (Texteditor)

≈ 420.000 LOC

Mozilla Thunderbird

≈ 900.000 LOC

Mozilla Firefox

≈ 37.000.000 LOC

Ford F150 pickup (2016)

≈ 150.000.000 LOC

# Lernziele

- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken

- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken

Fünf Phasen:

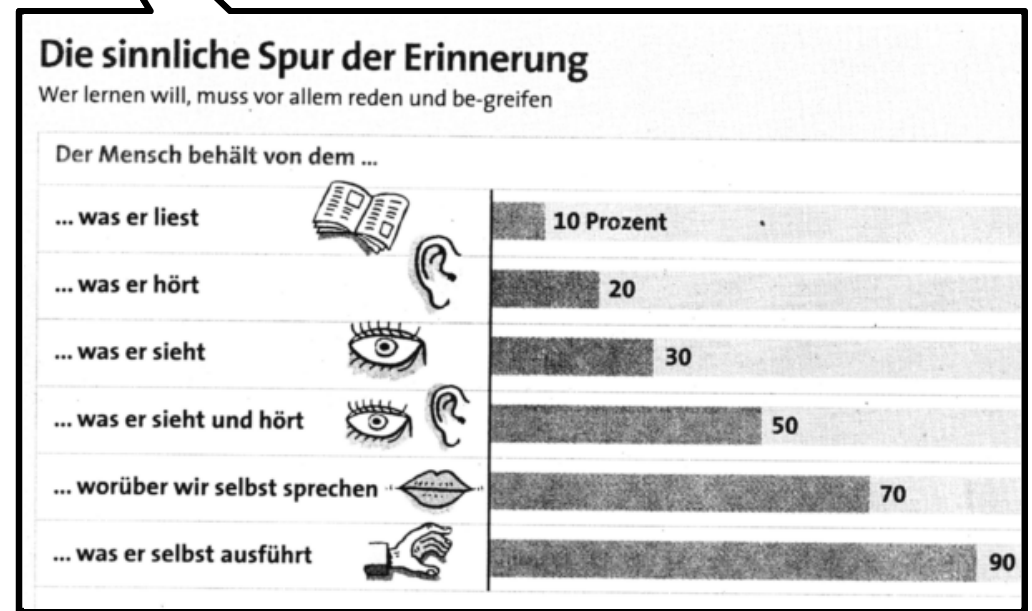
- Pflichtenheft
- Entwurf
- Implementierung
- Testphase
- Systemabnahme

# Lernziele

- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken
  
- Sammeln von praktischer Erfahrung bei der Planung von Software und in der Qualitätssicherung

# Lernziele

- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken
- Sammeln von praktischer Erfahrung bei der Planung von Software und in der Qualitätssicherung

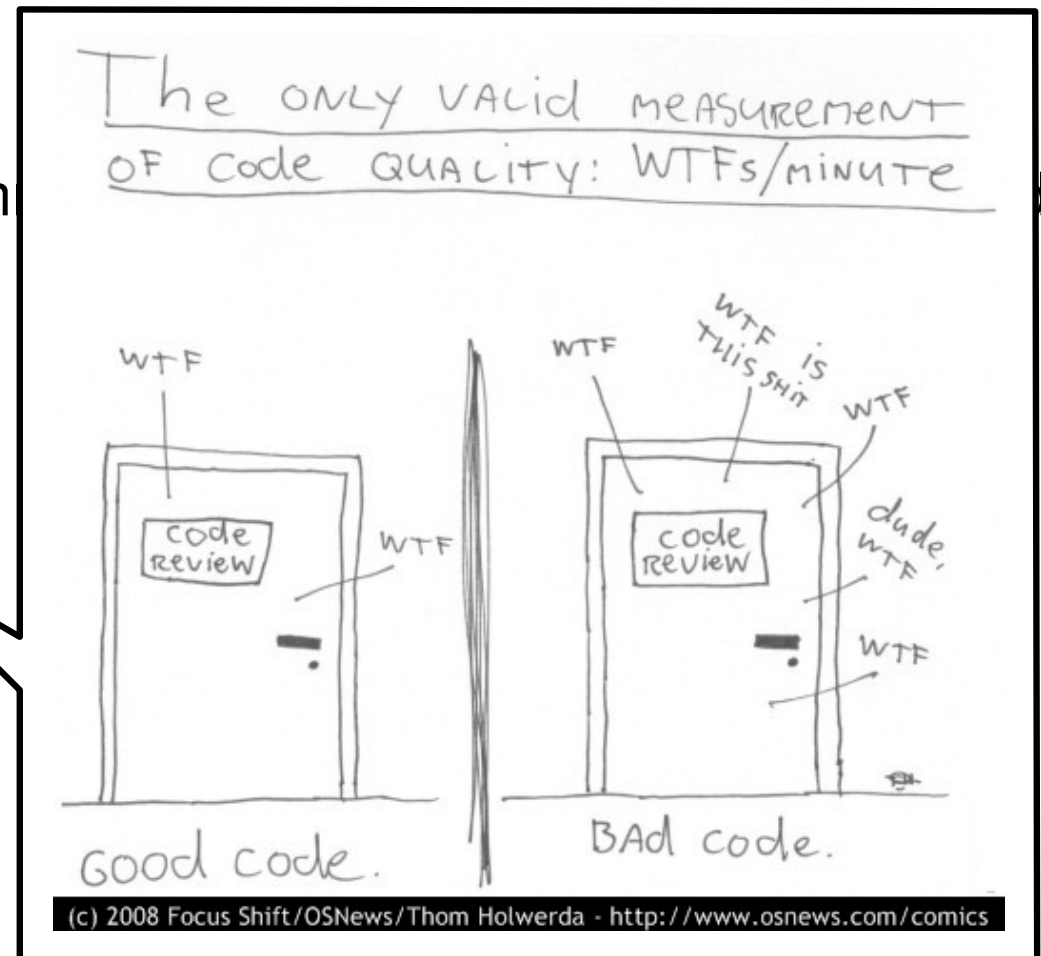




# Lernziele

- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken
- Sammeln von praktischer Erfahrung bei der Planung von Software und in der Qualitätssicherung
- Implementierungskompetenz

- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken
- Sammeln von praktischer Erfahrung in der Qualitätssicherung
- Implementierungskompetenz



# Lernziele

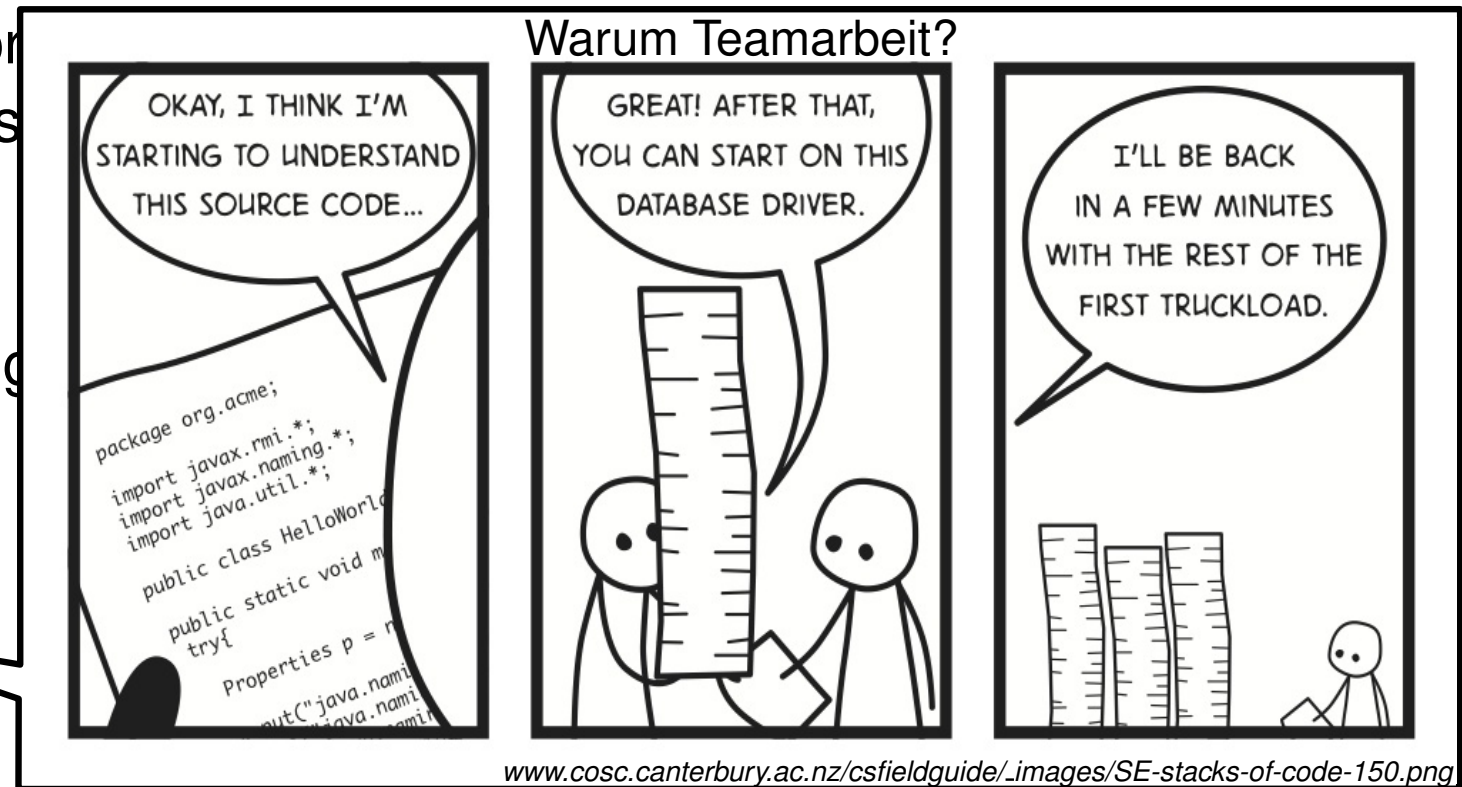
- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken
- Sammeln von praktischer Erfahrung bei der Planung von Software und in der Qualitätssicherung
- Implementierungskompetenz
- Teamarbeit

- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken

- Sammeln von pr...  
in der Qualitätss...

- Implementierung

- Teamarbeit



# Lernziele

- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken
- Sammeln von praktischer Erfahrung bei der Planung von Software und in der Qualitätssicherung
- Implementierungskompetenz
- Teamarbeit
- Präsentation

# Lernziele

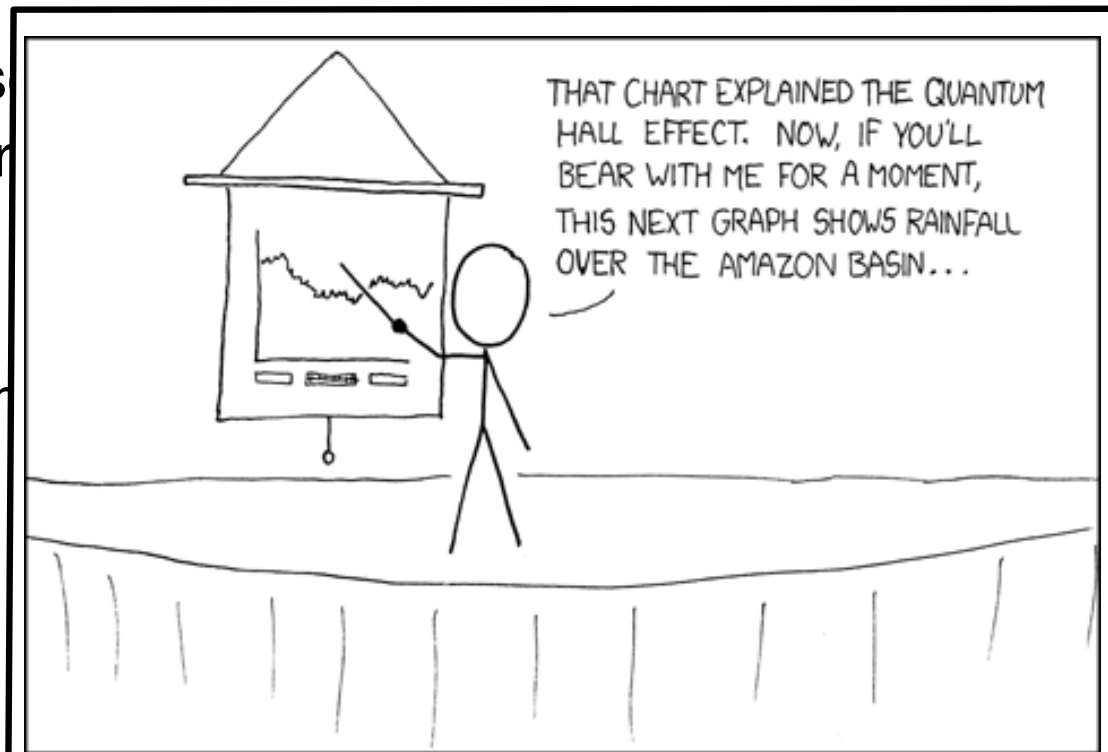
- Realisierung eines kompletten Softwareprojekts unter Benutzung von Softwareentwicklungstechniken

- Sammeln von praktischen Erfahrungen in der Qualitätssicherung

- Implementierungskonzepte

- Teamarbeit

- Präsentation



IF YOU KEEP SAYING "BEAR WITH ME FOR A MOMENT", PEOPLE TAKE A WHILE TO FIGURE OUT THAT YOU'RE JUST SHOWING THEM RANDOM SLIDES.

Quelle: <http://xkcd.com/365/>

# Anforderungen

- Aktiver Beitrag und aktive Teilnahme in jeder Phase von jedem

# Anforderungen

- Aktiver Beitrag und aktive Teilnahme in jeder Phase von jedem
- Teilnahme an den wöchentlichen Treffen



# Anforderungen

- Aktiver Beitrag und aktive Teilnahme in jeder Phase von jedem
- Teilnahme an den wöchentlichen Treffen
- Alle Dokumente müssen pünktlich (als PDF) abgegeben werden.
- Einen Tag vor jedem (wöchentlichen) Treffen: Einreichung des aktuellen Konzepts

# Anforderungen

- Aktiver Beitrag und aktive Teilnahme in jeder Phase von jedem
- Teilnahme an den wöchentlichen Treffen
- Alle Dokumente müssen pünktlich (als PDF) abgegeben werden.
- Einen Tag vor jedem (wöchentlichen) Treffen: Einreichung des aktuellen Konzepts
- Kolloquium nach jeder Phase
  - Präsentation (Ergebnisse der Phase) + Prüfungsgespräch

# Anforderungen

- Aktiver Beitrag und aktive Teilnahme in jeder Phase von jedem
- Teilnahme an den wöchentlichen Treffen
- Alle Dokumente müssen pünktlich (als PDF) abgegeben werden.
- Einen Tag vor jedem (wöchentlichen) Treffen: Einreichung des aktuellen Konzepts
- Kolloquium nach jeder Phase
  - Präsentation (Ergebnisse der Phase) + Prüfungsgespräch
- Die Note setzt sich zusammen aus
  - Qualität der vorgelegten Dokumente
  - Kolloquium
  - Qualität des Projektes

- Mailingliste: [i11-pse-betreuer@lists.kit.edu](mailto:i11-pse-betreuer@lists.kit.edu)
- Bei Fragen und Problemen, zur Abgabe von Dokumenten (max. 5 MB)
- Alternativ: direkte E-Mail an alle (!) Betreuer
- MS Teams – 2 Teams, sehen alle die Teams?
- Alle Treffen, Präsentationen und Prüfungsgespräche via MS Teams

- **Unmittelbarkeitsgrundsatz:** Es muss eine eigene, unmittelbare und vollständige Kenntnisnahme des gesamten Prüfungsgeschehens von allen Prüferinnen und Prüfern gewährleistet sein.
- Die Prüfung darf nur als Videoübertragung stattfinden, damit Prüfer/innen, Beisitzer/innen und Prüfling sich gegenseitig jederzeit sehen und hören können. Insbesondere ist sicherzustellen, dass der Prüfling sämtliche Prüferinnen und Prüfer zu jeder Zeit sehen kann (nicht nur die Prüferin bzw. den Prüfer, die oder der aktuell Fragen stellt) sowie sämtliche Prüferinnen und Prüfer den Prüfling.
- Also kurz: alle dauerhaft Mikrofon und Kamera an
- Nur auf freiwilliger Basis – Datenschutzerklärung
- Aber: ohne Zustimmung dieses Semester vermutlich keine Teilnahme an PSE im 5er-Team möglich

- Ohne Prüfung kein Bestehen
- **Aber:** Leistung in der Phase bestimmt primär die Note, hier lediglich Präsentation der Leistung
- Rücksicht auf besondere Situation
- Abmeldung vermutlich bis vor Prüfung möglich, dann aber Nachprüfung
- Technische Probleme kein Problem wenn “unverdächtig”
- Genau gleiches Setup wie wöchentliche Treffen

# Rückkehr zur Normalität?

- Derzeit nicht absehbar
- Aktuell keine persönlichen Treffen möglich
- Wir können (vermutlich) besser als andere auf persönliche Zusammentreffen verzichten und sollten es deshalb auch tun
- Wenn alles funktioniert gibt es keine persönlichen Treffen, zumindest solange die KIT-Vorgaben Prüfungen per Videokonferenz zulassen

- Testen und Feedback
- Unbedingt nötig: Bildschirm Teilen – notfalls mit Chrome Tab teilen
- Ton: Kopfhörer gegen Echo, aber auf Rauschen achten bei Headset
- Bildqualität: Licht von vorne hilft – z.B. eine einfache Lampe
- Netzkabel statt WLAN
- Fallback – z.B. via Handy



1. Organisation

2. Eure Aufgaben

3. Werkzeuge



# Campus-Routenplaner

From:

To:

Get directions





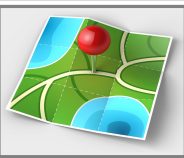
# Campus-Routenplaner

From: **AUDIMAX**

To:

Get directions





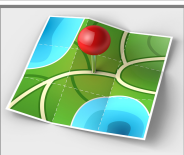
# Campus-Routenplaner

From: **AUDIMAX**

To: **50.34**

Get directions





# Campus-Routenplaner

From: **AUDIMAX**

To: **50.34**

Get directions



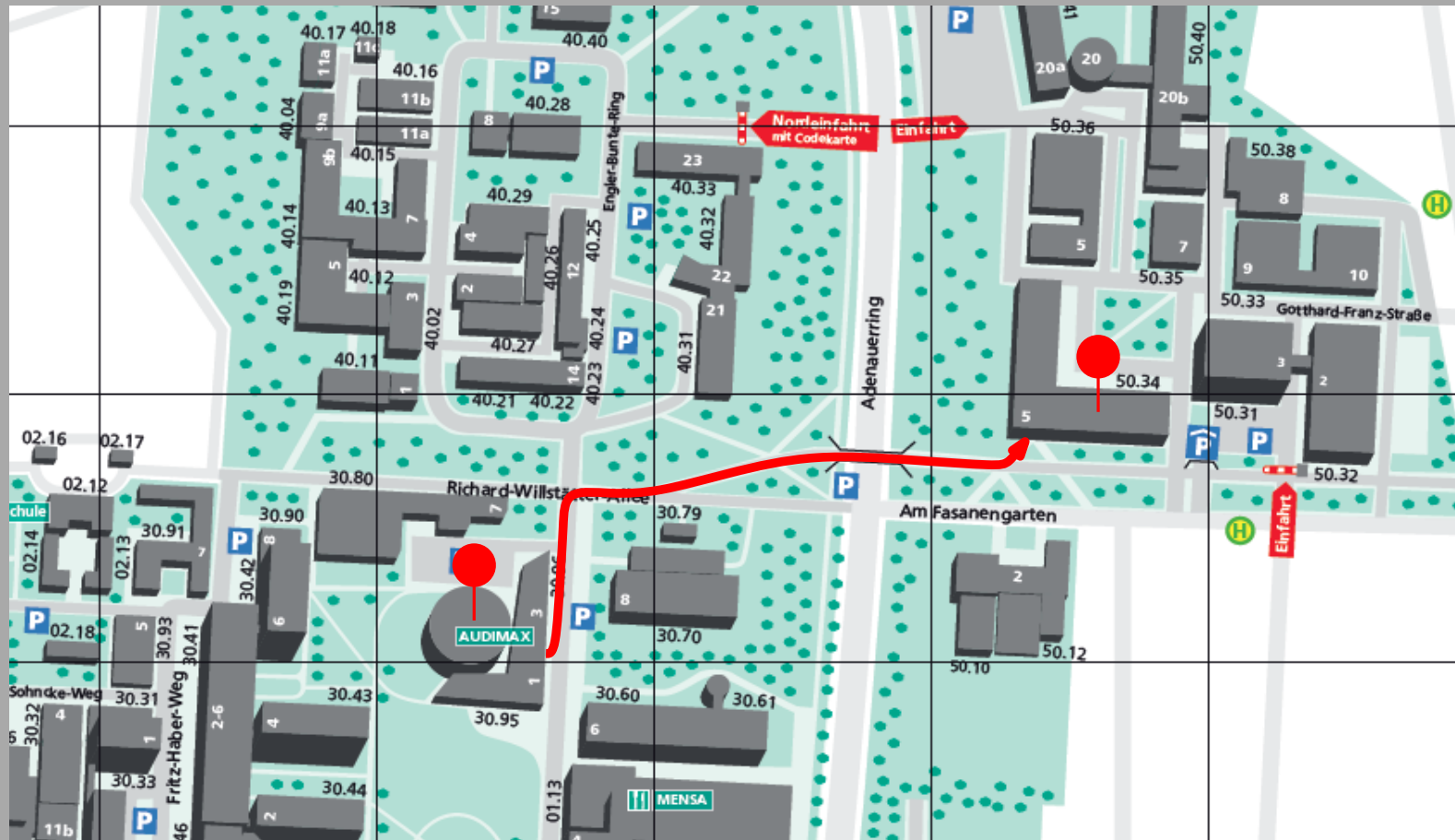


# Campus-Routenplaner

From: **AUDIMAX**

To: **50.34**

Get directions



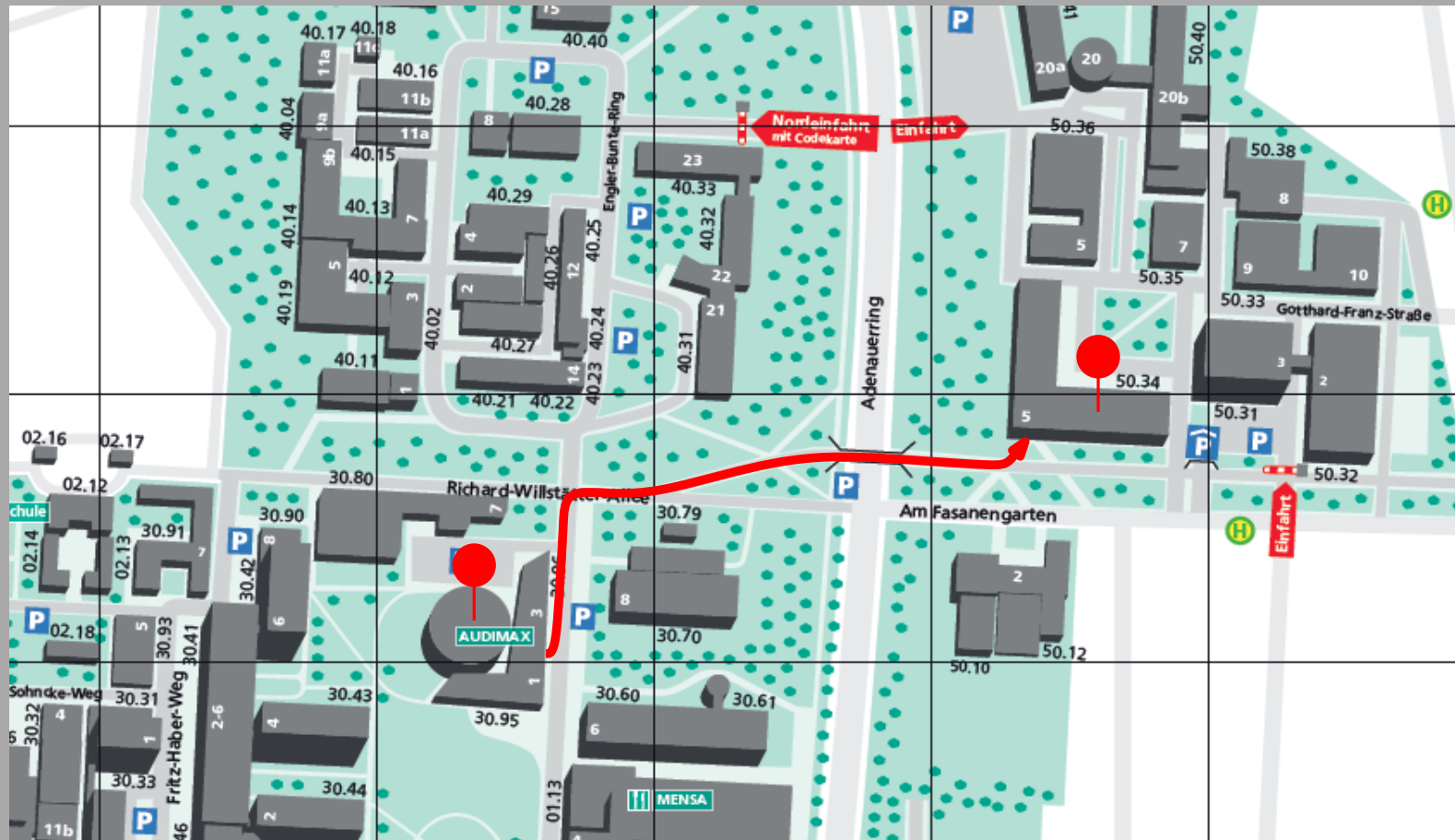


# Campus-Routenplaner

From: **AUDIMAX**

To: **50.34, Raum 315**

Get directions



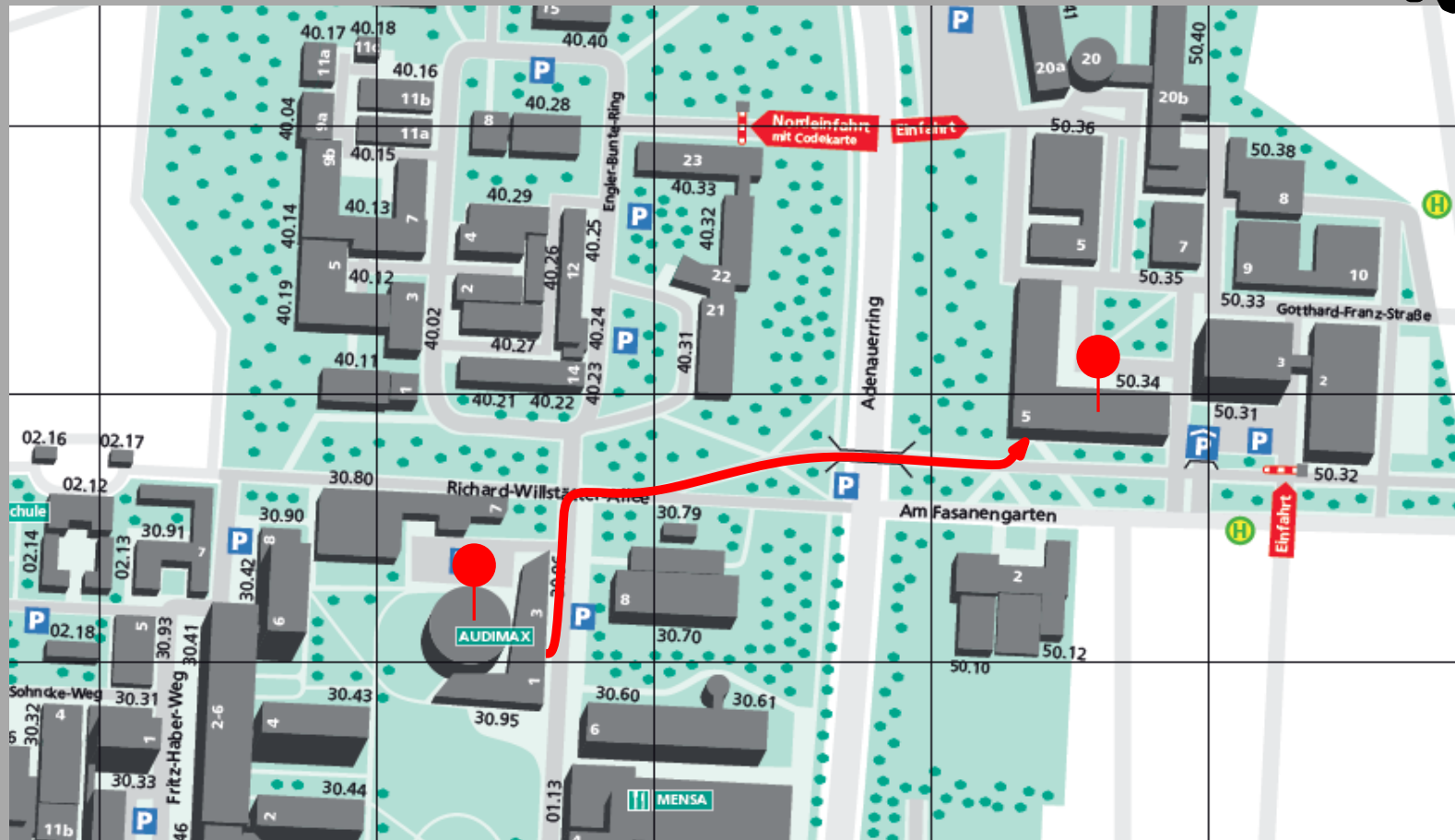


# Campus-Routenplaner

From: **AUDIMAX**

To: **50.34, Raum 315**

Get directions 





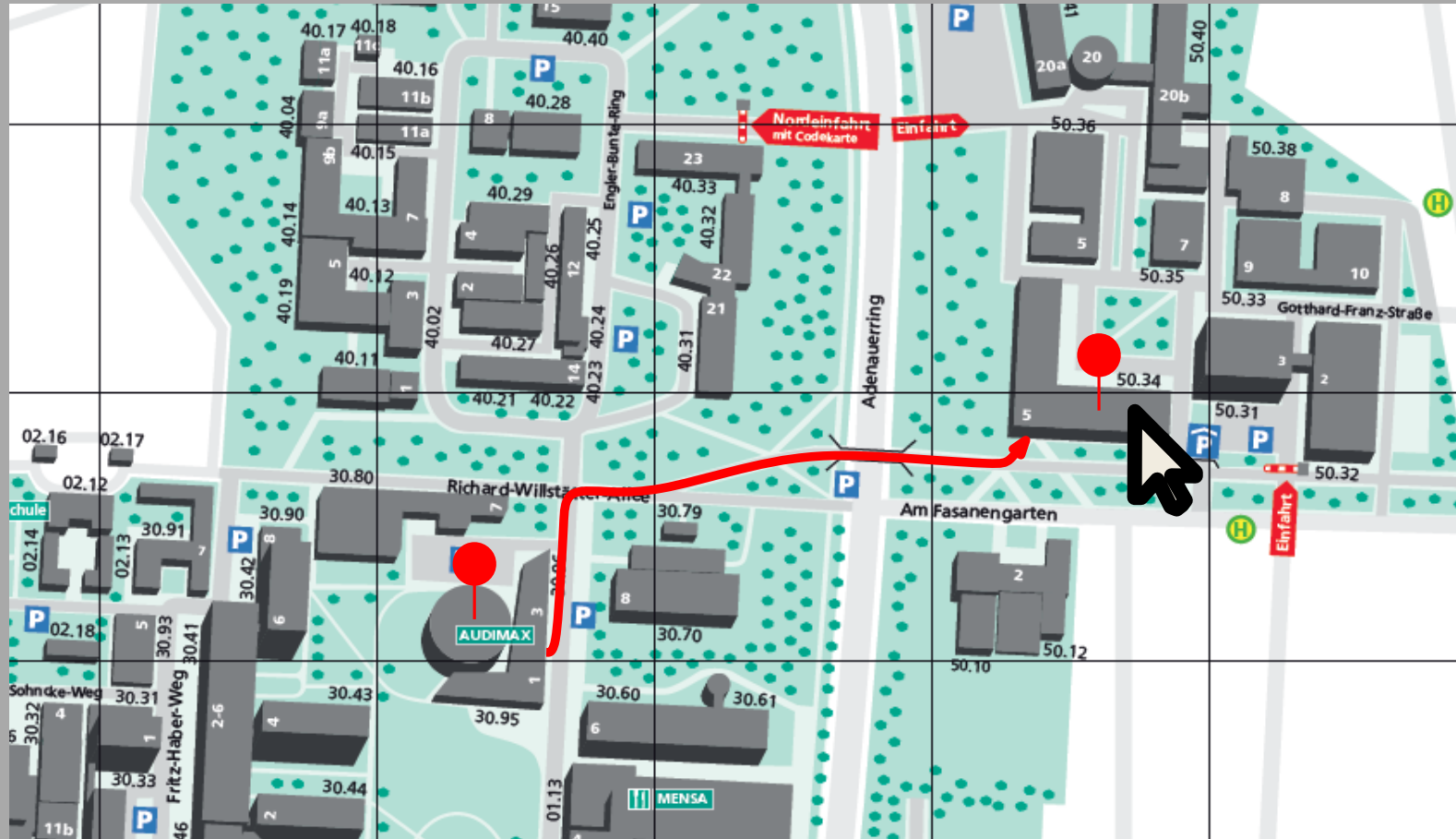


# Campus-Routenplaner

From: **AUDIMAX**

To: **50.34, Raum 315**

Get directions





# Campus-Routenplaner

From: **AUDIMAX**      To: **50.34, Raum 315**      [Get directions](#)

**Etage 0**

The floor plan shows a route starting from the entrance (Eingang) on the left, passing through a staircase (Treppe) and a BMZ area, and ending in room 001a. The route is highlighted in red. The plan includes rooms 009-004, 037-040, 034-036, 041, 046, 001c, 001b, 002, and 001a. A staircase (Treppe) and a BMZ (Büro-Möbelzone) are also labeled.



# Campus-Routenplaner

From: **AUDIMAX**      To: **50.34, Raum 315**      [Get directions](#)

**Etage 0**

The floor plan shows a route starting from the entrance (Eingang) on the left, passing through a staircase (Treppe) area, and ending in room 001a. The route is highlighted in red. Other rooms and areas include 001b, 001c, 002, 004, 005, 006, 007, 008, 009, 034, 035, 036, 037, 038, 039, 040, 041, and 046. A BMZ (Büro-Möbelzone) is also marked.



# Campus-Routenplaner

From: **AUDIMAX**      To: **50.34, Raum 315**      [Get directions](#)

**Etage 0**

- Etage -2
- Etage -1
- Etage 0
- Etage 1
- Etage 2
- Etage 3



# Campus-Routenplaner

From: **AUDIMAX**      To: **50.34, Raum 315**      [Get directions](#)

**Etage 0**

- Etage -2
- Etage -1
- Etage 0
- Etage 1
- Etage 2
- Etage 3



# Campus-Routenplaner

From: **AUDIMAX**      To: **50.34, Raum 315**      [Get directions](#)

**Etage 3**

The floor plan shows a grid of rooms. The starting point is a staircase labeled 'Treppe' in room 332. The route proceeds through rooms 331, 301, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, and finally reaches room 315. The route is highlighted in red. There are also 'Fluchtbalkon' (fire balcony) areas shown.

# Campus-Routenplaner – Aufgabe

## Entwurf und Implementierung eines Routenplaners für den KIT-Campus

- Einfache Suche nach Start und Ziel
- Anzeige des kürzesten Pfades
- Dijkstras Algorithmus für das Routing
- Route von und zu **Gebäuden**
- Routenberechnung innerhalb von Gebäuden



## Administrationswerkzeug

- Karte hinzufügen (Hintergrundbild)
- Hinzufügen (und auch Löschen) von Kanten (Wege) und Knoten (Kreuzungen)
- Hinzufügen/Löschen von Gebäuden und Hinzufügen von Informationen zu diesen Gebäuden



# Campus-Routenplaner – Erweiterungen

- Verschiedene Metriken für das Routing
- Verschiedene Routen
- Neue Kanten-/Knotenattribute
- Routing zum nächstgelegenen Eingang
- Straßen temporär als gesperrt markieren
- Kürzeste Wege können durch ein Gebäude gehen
- Schöne Darstellung einer Route innerhalb eines Gebäudes
- ...



# Bis zur nächsten Woche...

- Kennen lernen, Videokonferenz testen
- Umgang mit git &  $\text{\LaTeX}$  lernen
- Einarbeitung in die anderen Werkzeuge, wie Lucidchart, Eclipse, ...
- Lesen der Aufgabe!
- Wählen der Teamleiter für die verschiedenen Phasen
- Über Struktur und besondere Funktionalitäten von eurem System diskutieren und aufschreiben (Pflichtenheft)

1. Organisation

2. Eure Aufgaben

3. Werkzeuge

# Werkzeuge

## Git (oder eine andere Versionskontrolle)

- Es ist **verpflichtend** zu verwenden!
- Repository bei `https://git.scc.kit.edu` mit Zugriff für Betreuer

# Werkzeuge

## Git (oder eine andere Versionskontrolle)

- Es ist **verpflichtend** zu verwenden!
- Repository bei `https://git.scc.kit.edu` mit Zugriff für Betreuer

## L<sup>A</sup>T<sub>E</sub>X

- Wir empfehlen L<sup>A</sup>T<sub>E</sub>X für die Dokumentation
- Overleaf `http://www.scc.kit.edu/dienste/13330.php`

# Werkzeuge

## Git (oder eine andere Versionskontrolle)

- Es ist **verpflichtend** zu verwenden!
- Repository bei `https://git.scc.kit.edu` mit Zugriff für Betreuer

## L<sup>A</sup>T<sub>E</sub>X

- Wir empfehlen L<sup>A</sup>T<sub>E</sub>X für die Dokumentation
- Overleaf `http://www.scc.kit.edu/dienste/13330.php`

## Abbildungen & Präsentationen

- Wir verwenden hauptsächlich Ipe (`http://ipe.otfried.org/`)

## Git (oder eine andere Versionskontrolle)

- Es ist **verpflichtend** zu verwenden!
- Repository bei `https://git.scc.kit.edu` mit Zugriff für Betreuer

## L<sup>A</sup>T<sub>E</sub>X

- Wir empfehlen L<sup>A</sup>T<sub>E</sub>X für die Dokumentation
- Overleaf `http://www.scc.kit.edu/dienste/13330.php`

## Abbildungen & Präsentationen

- Wir verwenden hauptsächlich Ipe (`http://ipe.otfried.org/`)

## UML

- Lucidchart (`https://www.lucidchart.com/pages/education`)
- MS Visio (`http://kit.onthehub.com`)
- yED (`https://www.yworks.com/products/yed`)
- Dia (`https://wiki.gnome.org/Apps/Dia`), ...

# Werkzeuge

## Projektplanung

- Planner (<https://wiki.gnome.org/Apps/Planner>), ...

# Werkzeuge

## Projektplanung

- Planner (<https://wiki.gnome.org/Apps/Planner>), ...

## Programmieren

- Eclipse
- NetBeans, ...



## Projektplanung

- Planner (<https://wiki.gnome.org/Apps/Planner>), ...

## Programmieren

- Eclipse
- NetBeans, ...

## Validierung

- Automatisiertes Testen: JUnit, SoapUI, ...
- Automatisiertes Messen der Unit Test Qualität (Mutation Tests): ...
- Java Programm Abdeckungswerkzeuge (engl. code coverage tools): Clover, Cobertura, JaCoCo, EMMA, ...

# Programmiersprache

- Java + JavaFX (Desktopanwendung)

- Java + JavaFX (Desktopanwendung)
- Alternativ: C++(11/14/17) (+ Qt), Android-App, Webanwendung, . . .

- Java + JavaFX (Desktopanwendung)
- Alternativ: C++(11/14/17) (+ Qt), Android-App, Webanwendung, . . .

Alternativen nur:

- In Absprache mit uns
- Mit Vorwissen/eigener Einarbeitung
- Wenn alle einverstanden sind

# Wann?

# Fragen

