



# Codegeneration with AI

# Gliederung

- 1) Was ist KI - Definition (Felix)
- 2) Funktionsweise + Umsetzung von AI\*s (Mike)
- 3) TabNine + Live Beispiel: GitHub AutoPilot (Sami)
- 4) Ausblick und Fazit (Johannes)



# Was ist KI?

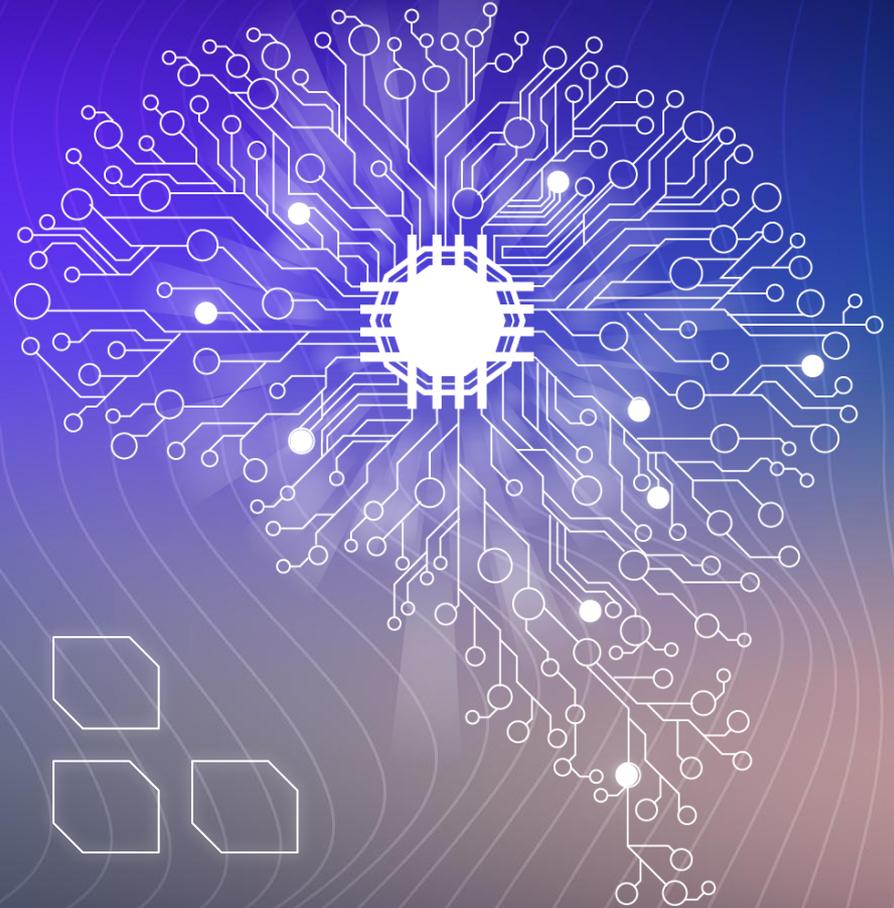
# Definition

„Künstliche Intelligenz ist die Eigenschaft eines IT-Systems, »  
mensenähnliche«, intelligente Verhaltensweisen zu zeigen“

- Bitkom e.V. und Deutsches Forschungszentrum für künstliche Intelligenz

# Kernfähigkeiten

- Wahrnehmen
- Verstehen
- Handeln
- Lernen



# Schwache KI

- Beschäftigt sich mit konkreten Anwendungsproblemen
- Simuliert nur die Intelligenz
- Den Menschen intelligent beim Erreichen seiner Ziele zu unterstützen
- Beispiel: Navigationssysteme, Spracherkennung, Korrekturvorschläge bei Suchen, Codegenerierung GitHub Copilot

# Starke KI

- Erschaffung einer allgemeinen Intelligenz, die der Menschen gleicht oder diese übertrifft
- IT-Systeme sind in der Lage menschenähnlich zu denken und zu handeln
- Entwickelt sich selbst weiter und lernt aus Fehlern

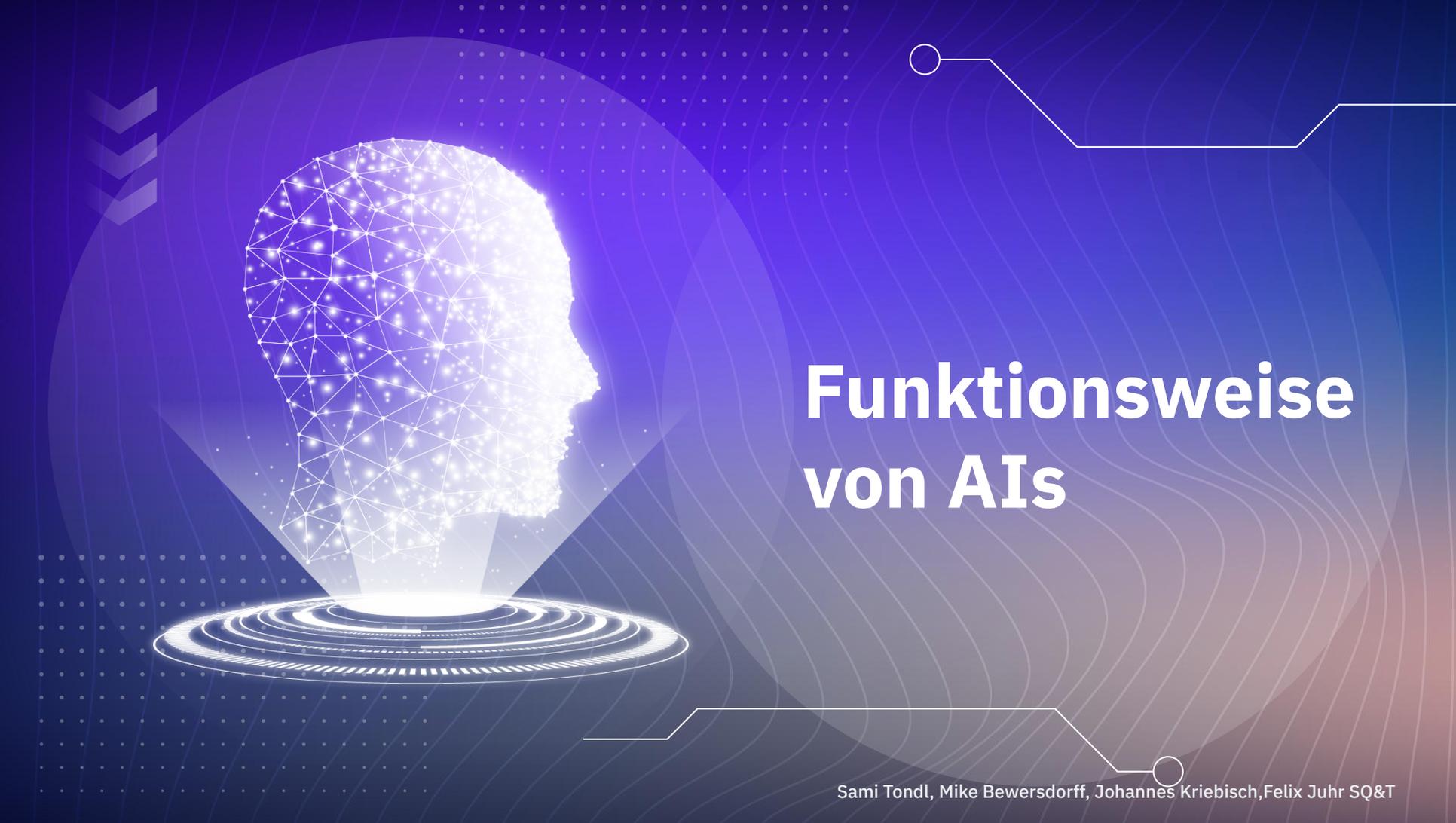


# Fähigkeiten einer wirklichen Intelligenz

- Logisches Denken
- Treffen von Entscheidungen bei Unsicherheit
- Planen
- Kommunikation in natürlicher Sprache

# Ziele der KI

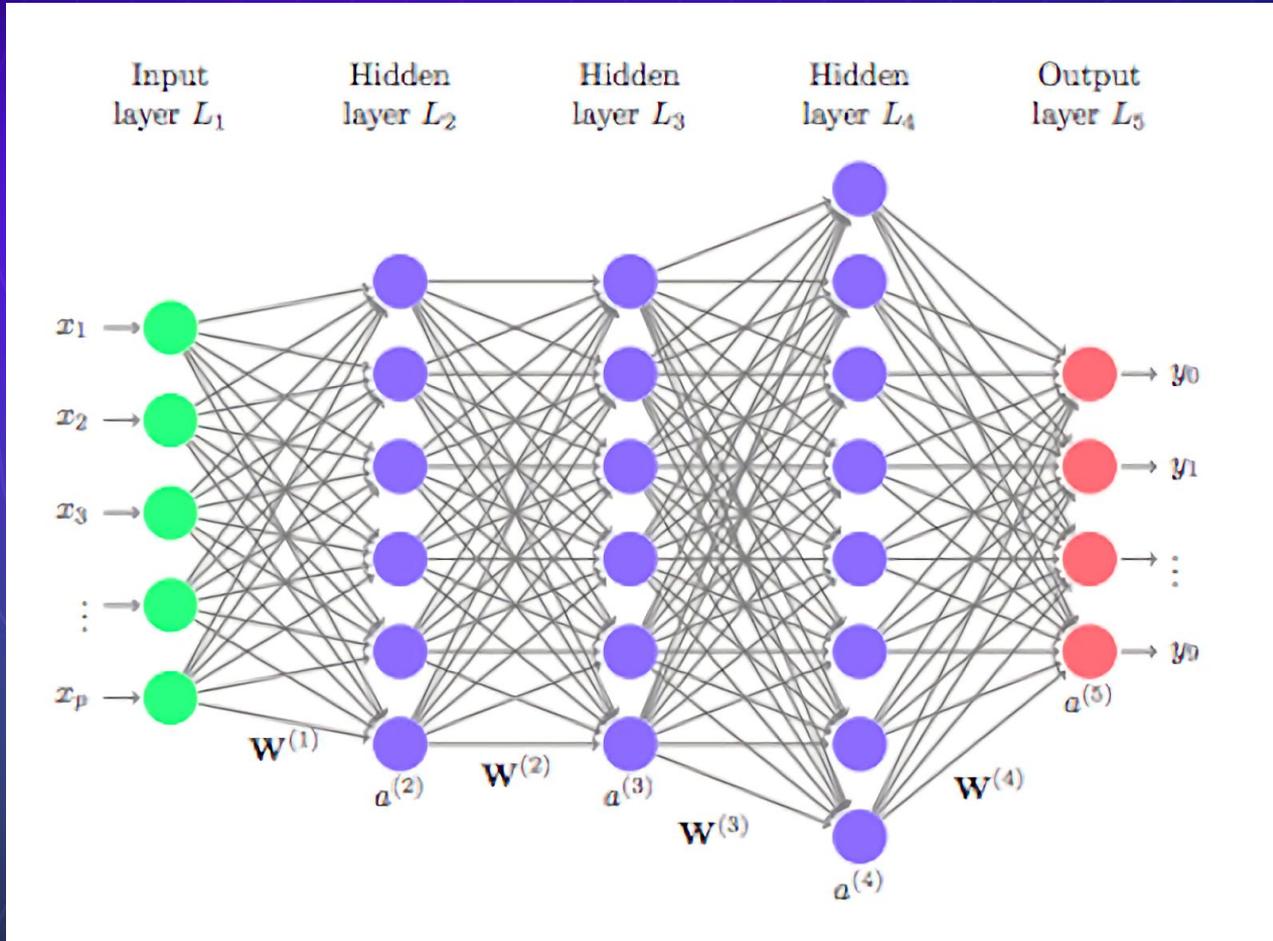
- Menschen unterstützen
- Unangenehme Arbeit deutlich zu vermeiden
- Den Menschen nicht überflüssig zu machen



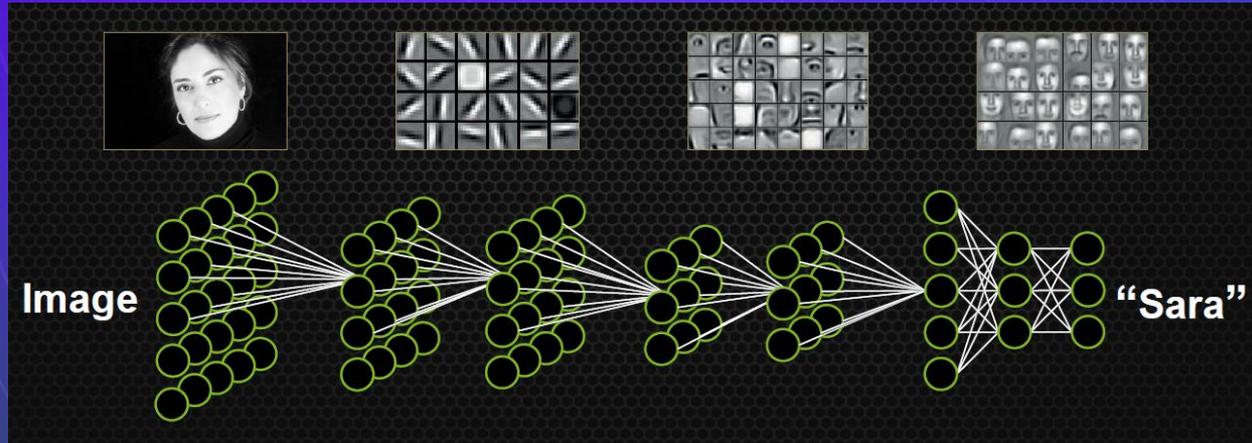
# Funktionsweise von AIs

# Umsetzung von AIs

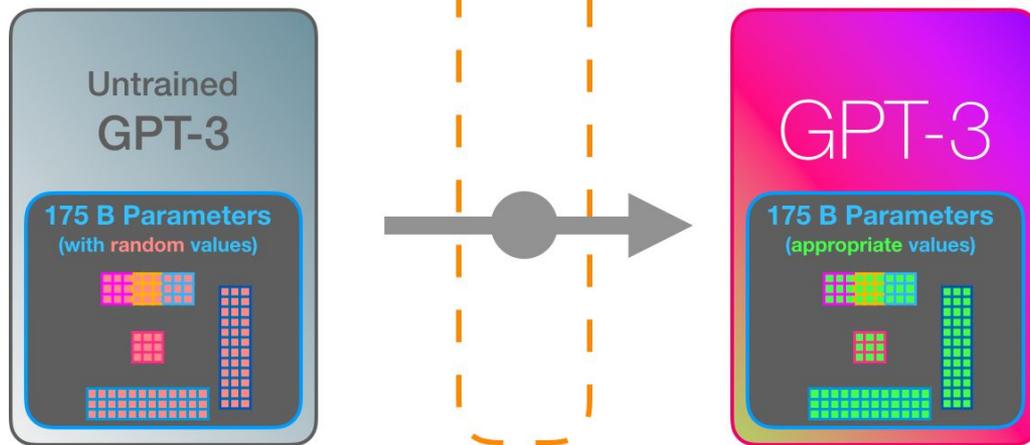
- Jedes System das auf einen Input einen “intelligenten” Output zurück gibt
- Neuronale Netzwerke am erfolgreichsten
- Meist automatisch via Punktesystem trainiert
- Qualität des Outputs skaliert mit Größe des Trainingssets & der Weights/Parameter



# Erkennen komplexer Strukturen



### Unsupervised Pre-training





tabnine

# TabNine

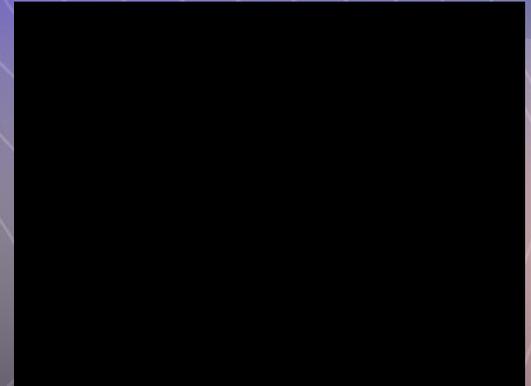
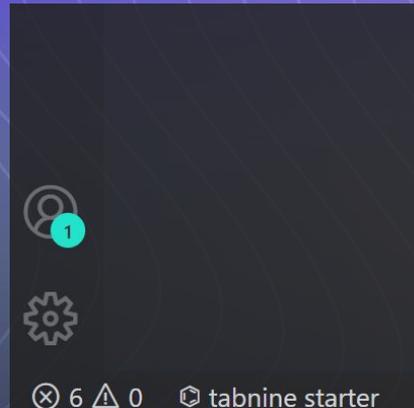
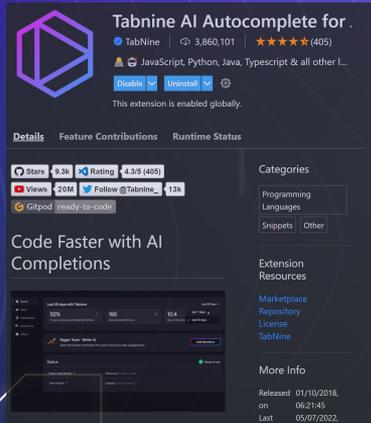
AI Autocomplete

# Installation

Install  
in  
Vscode

Wait for  
AI build

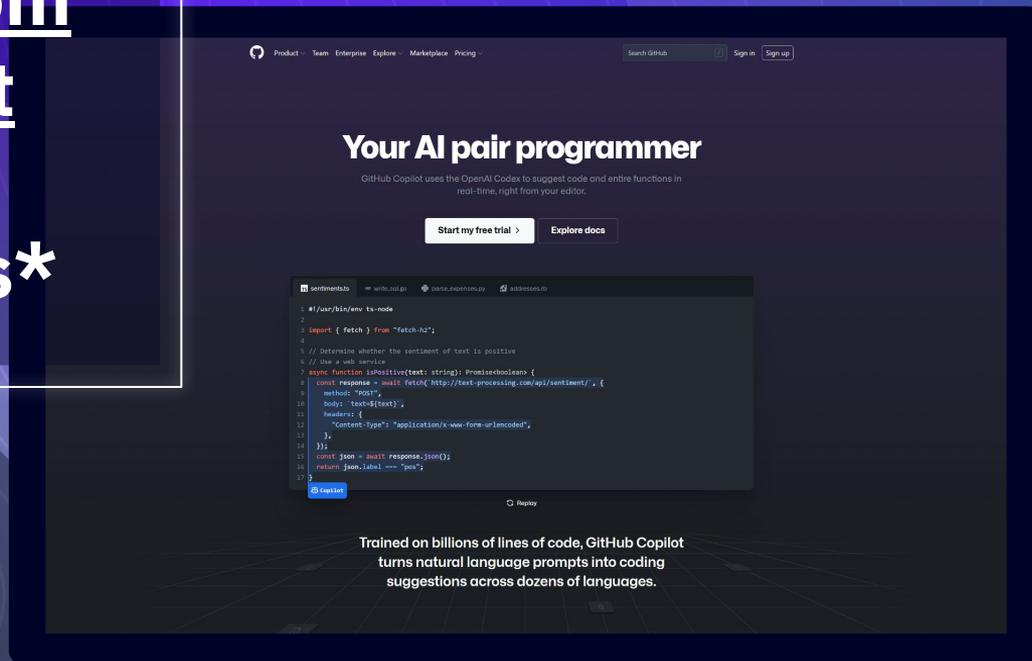
Try





## Beispiel

[https://github.com](https://github.com/features/copilot)  
[/features/copilot](https://github.com/features/copilot)  
10\$ per Month -  
Free for students\*



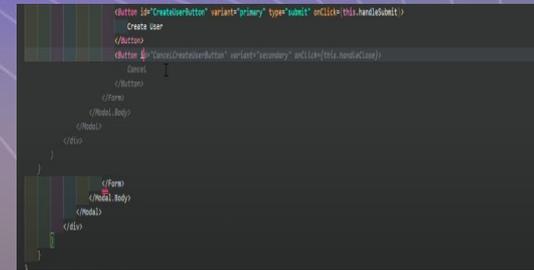
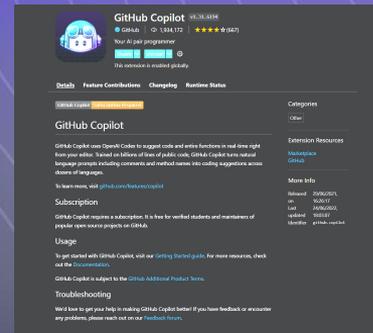
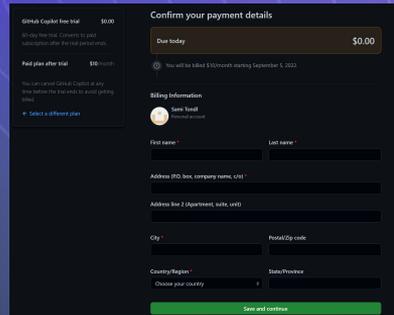
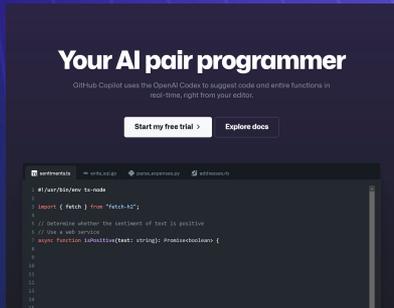
# Installation

Sign In

Go through billing\*

Install in Vscode

Try



\* Github hat kostenlose Studentenversionen angekündigt

# HOW-TO

Erste hints  
eintippen  
z.B.  
Funktion's  
Name

Zum  
Akzeptieren  
Tabulator  
drücken

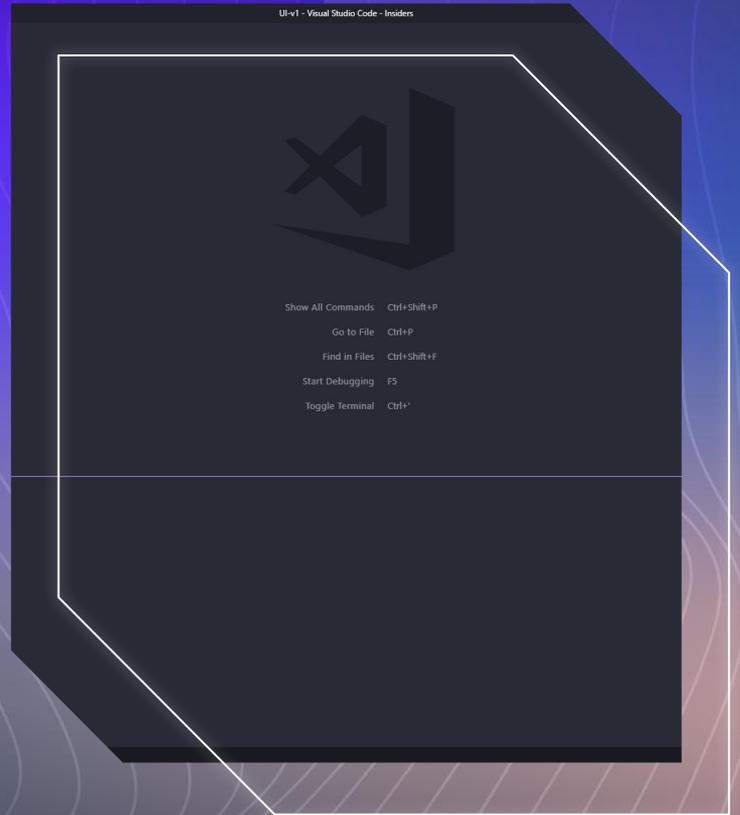
```
File Edit Selection View Go Run Terminal Help
CreateUserWidget.js - ANSERAPP - Visual Studio Code

EXPLORER
src > components > userManagement > CreateUserWidget > render
  Create User
  </Button>
  <Modal show={showUserCreateDialog} onHide={this.handleClose}>
    <Modal.Header closeButton>
      Create User
    </Modal.Header>
    <Modal.Body>
      <Form>
        <Form.Group className="mb-3">
          <Form.Label>UserID</Form.Label>
          <Form.Control id="UserIDInput" name="UserID" type="text" placeholder="UserID" onChange={this.handleChange} />
        </Form.Group>
        <Form.Group className="mb-3">
          <Form.Label>Username</Form.Label>
          <Form.Control id="UserNameInput" name="userName" type="text" placeholder="Username" onChange={this.handleChange} />
        </Form.Group>
        <Form.Group className="mb-3">
          <Form.Label>Password</Form.Label>
          <Form.Control id="PasswordInput" name="password" type="password" placeholder="Password" onChange={this.handleChange} />
        </Form.Group>
        <Form.Group className="mb-3">
          <Form.Label>Ist Admin</Form.Label>
          <Form.Check id="IsAdministratorInput" name="isAdministrator" type="checkbox" onChange={this.handleChange} />
        </Form.Group>
        <Button id="CreateUserButton" variant="primary" type="submit" onClick={this.handleSubmit}>
          Create User
        </Button>
      </Form>
    </Modal.Body>
  </Modal>
</div>
}
}

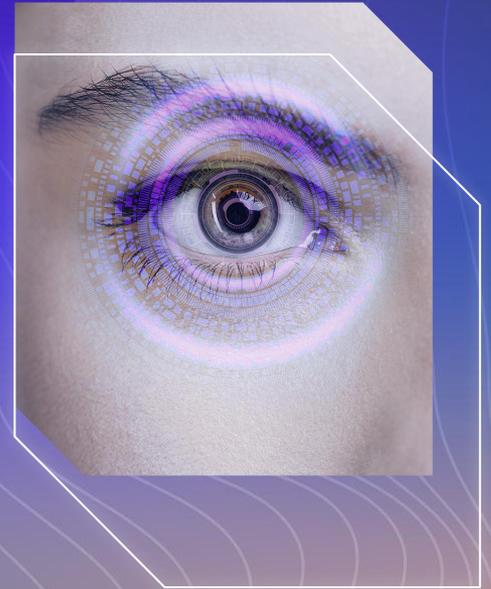
const mapDispatchToProps = dispatch => bindActionCreators({
  showUserCreateAction: userManagementActions.showUserCreateAction,
  hideUserCreateAction: userManagementActions.hideUserCreateAction,
  createUserAction: userManagementActions.createUserAction
}, dispatch)
```

## Beispiel

# Live Beispiel GitHub AutoPilot



# Ausblick und Fazit



# Technische Grenzen

- Fehleranfällig bei steigender Komplexität
- Begrenzt bei fachspezifischen Anwendungen
- Aktualität des verwendeten Codes
- Sicherheit der Programme
- Barriere der natürlichen Sprache



• • • • •

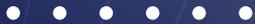
# Lizenzrecht des Codes

- **Kommerzielle Nutzung ist schwer**
- **Basiert der Code auf Open Source keine Kommerzielle Nutzung möglich**



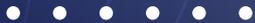
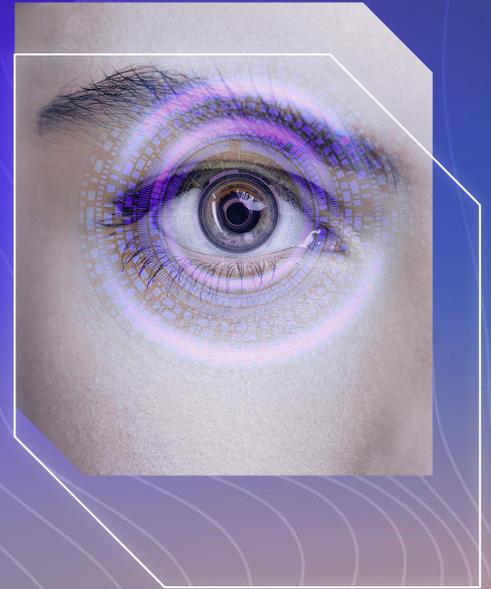
# Der Stand des Programmierers

- **Schwerpunktverlagerung in der Zukunft möglich**
- **Lesekompetenz löst die Schreibkompetenz ab**
- **Überwachung der AI-gestützten Programmierung**



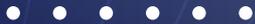
# Vereinfachter Einstieg in die Programmierung

- Ermöglicht Problemlösung für Menschen ohne Programmiererfahrung
- Syntax-Vorkenntnisse nicht zwingend erforderlich



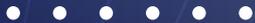
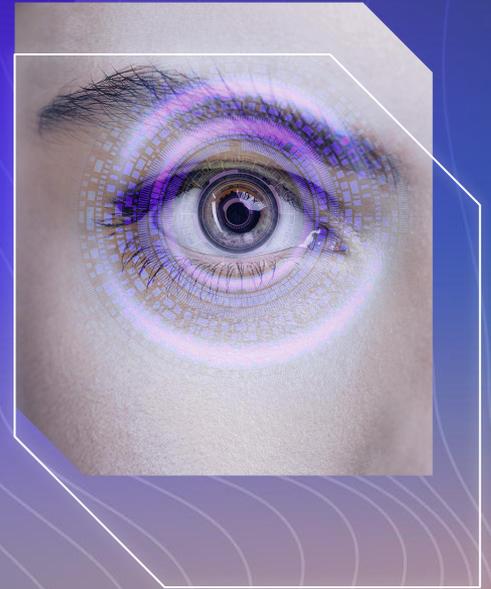
# Anwendung in der Kommerziellen Programmierung

- Entwicklungsprozess kann beschleunigt werden
- Verlagerung der Schwerpunkte auf Test und Qualität
- Entwicklung Komplexer Anwendungen wird vereinfacht



# Fazit

- **Enormes Potenzial für die Programmierung**
- **Senken von Kosten bei höherer Effizienz**



# Quellen

GitHub Copilot · Your AI pair programmer. (o. D.). GitHub. <https://github.com/features/copilot>  
AI Assistant for software developers | Tabnine. (o. D.). tabnine. <https://www.tabnine.com/>

## DESIGN:

Korean AI Agency Pitch Deck | Google Slides & PowerPoint. (o. D.). Slidesgo.  
<https://slidesgo.com/theme/korean-ai-agency-pitch-deck#search-ai&position-6&results-7>

<https://www.heise.de/hintergrund/Programmieren-und-KI-Kuenstliche-Intelligenz-in-der-Softw-areentwicklung-6150110.html>

[https://www.dfki.de/fileadmin/user\\_upload/import/9744\\_171012-KI-Gipfelpapier-online.pdf](https://www.dfki.de/fileadmin/user_upload/import/9744_171012-KI-Gipfelpapier-online.pdf)

[http://www.informatik.uni-oldenburg.de/~iug08/ki/Grundlagen\\_Starke\\_KI\\_vs.\\_Schwache\\_KI.html](http://www.informatik.uni-oldenburg.de/~iug08/ki/Grundlagen_Starke_KI_vs._Schwache_KI.html)

<https://www.heise.de/hintergrund/Programmieren-und-KI-Kuenstliche-Intelligenz-in-der-Softwareentwicklung-6150110.html>

<https://www.heise.de/ratgeber/Entwickeln-mit-KI-IntelliJ-IDEA-Visual-Studio-IntelliCode-und-Tabnine-im-Test-6150120.html>

<https://entwickler.de/programmierung/kunstliche-intelligenz-in-der-softwareentwicklung>

<https://nixsolutions-ai.com/polycoderopensourceai/>

<https://chatbots-und-ki.podigee.io/6-codegenerierung>

<https://www.ionos.de/digitalguide/websites/web-entwicklung/was-ist-low-code/#:~:text=Der%20Begriff%20Low%20Code%20bedeutet%20auf%20Deutsch%20so,aufw%C3%A4ndig%20programmiert%20werden%20muss.%20Low%20Code%20ist%20>