

Testumgebung

Melhorn Tobias, Hermüller Philipp, Pohn Patrick, Gnadlinger Johannes

Grundsätzliche Erklärung des Projektes

- Schüler überwachen**
 - Screenshot anzeigen
 - Zeilen im Code zählen
- Benachrichtigung bei möglichen Schummelversuch**
- ...

Teacher-GUI

Teacher Client

Optionen Student-Details Abgabe

Schülerliste

Port:

Angabe:

Speicherort (Screenshots, Abgaben):

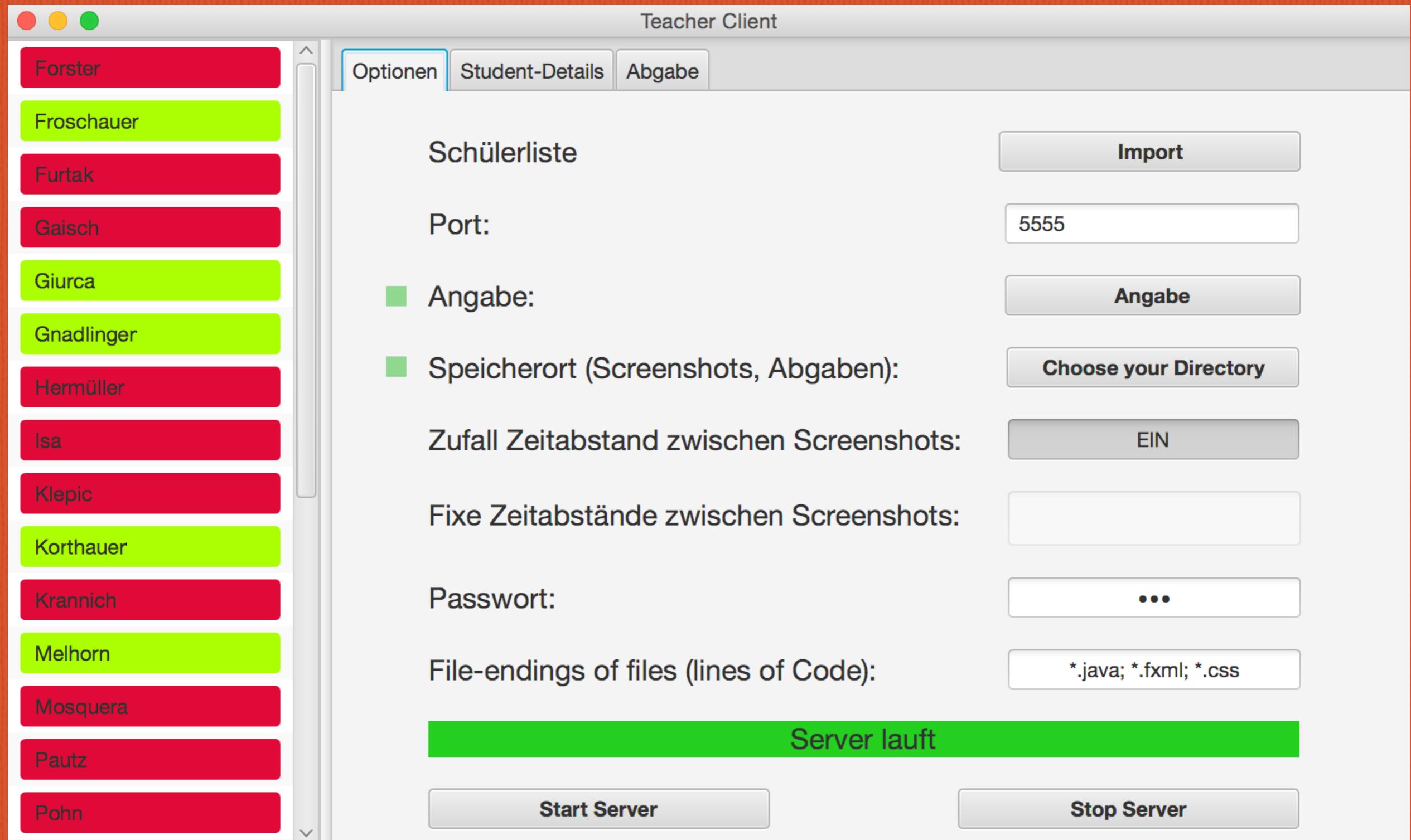
Zufall Zeitabstand zwischen Screenshots:

Fixe Zeitabstände zwischen Screenshots:

Passwort:

File-endings of files (lines of Code):

Teacher-GUI



importiert



leer

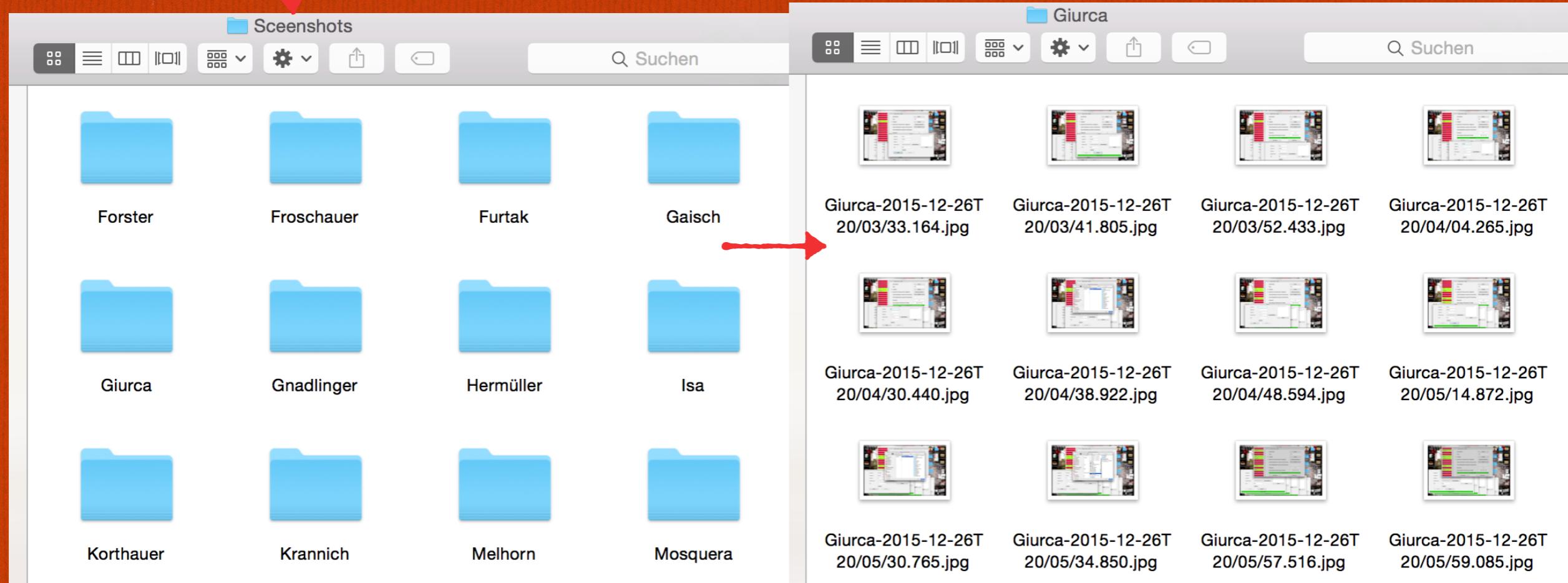
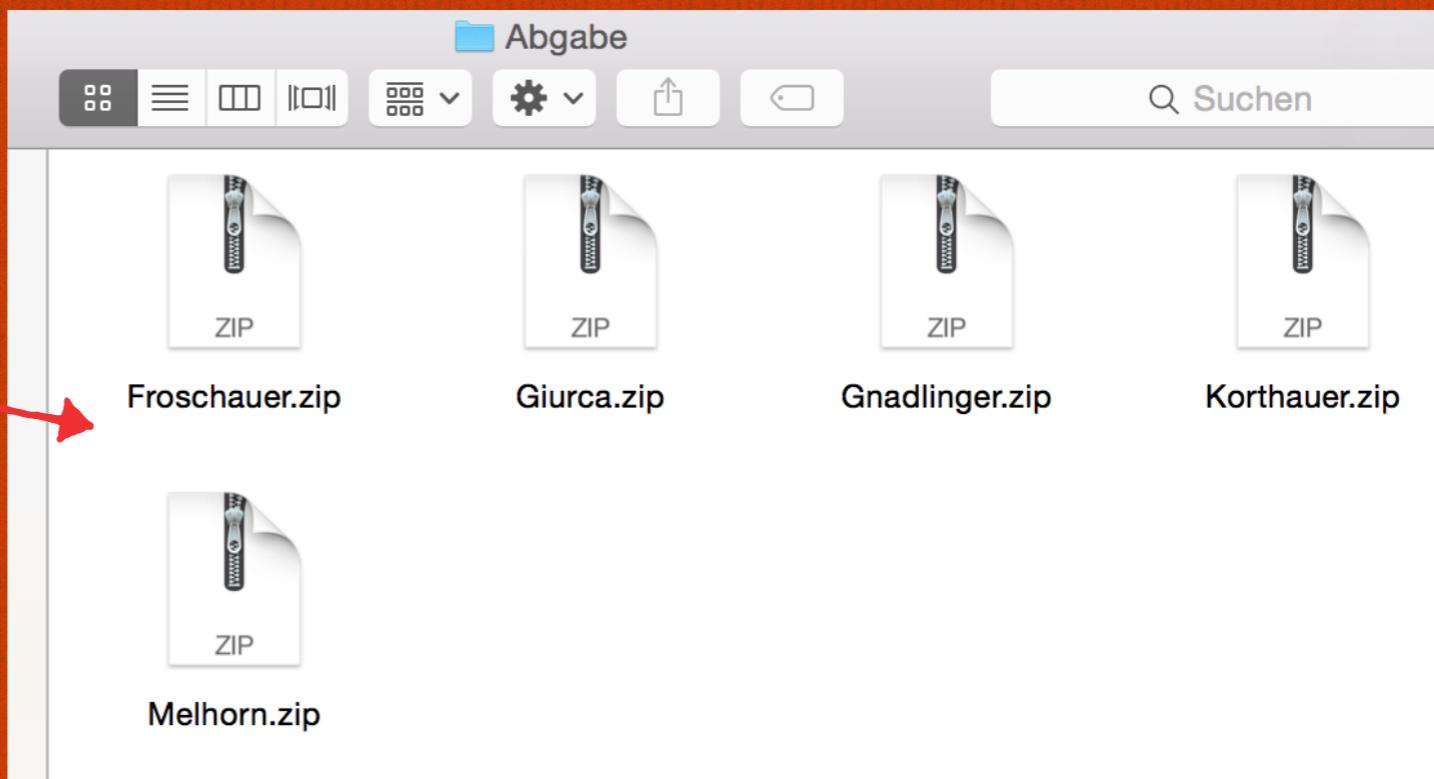
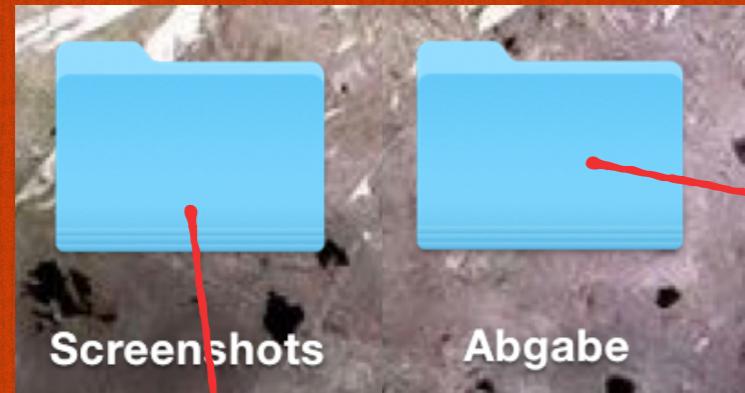
| |
|------------|
| Forster |
| Froschauer |
| Furtak |
| Gaisch |
| Giurca |
| Gnadlinger |
| Hermüller |
| Isa |
| Klepic |
| Korthauer |
| Krannich |
| Melhorn |
| Mosquera |
| Pautz |
| Pohn |

| |
|------------|
| Forster |
| Froschauer |
| Furtak |
| Gaisch |
| Giurca |
| Gnadlinger |
| Hermüller |
| Isa |
| Klepic |
| Korthauer |
| Krannich |
| Melhorn |
| Mosquera |
| Pautz |
| Pohn |

eingeloggt

| |
|--------------|
| Forster |
| Froschauer |
| Furtak |
| Gaisch |
| Giurca |
| Gnadlinger |
| Hermüller |
| Isa |
| Klepic |
| Korthauer |
| Krannich |
| Melhorn |
| Mosquera |
| Pautz |
| Pohn |
| Pühringer |
| Rudelsdorfer |
| Saiz |
| Steininger |
| Stelzmüller |
| Tanzer |

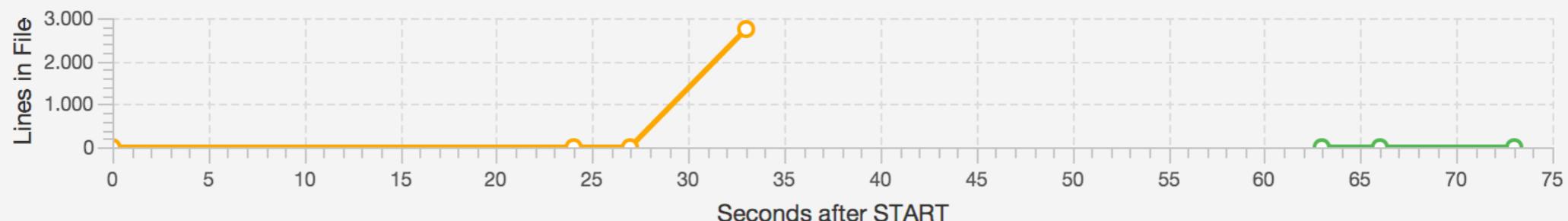
unerlaubt ausgeloggt



Teacher Client

Optionen Student-Details Abgabe

Lines of Code in Project



Project Structure:

```

TestumgebungNeu [RemoteControl] (~/Desktop)
  - .idea
  - out
  - src
    - main
      - java
        - at.htl.remotecontrol
          - actions
            - ClickMouse
            - MoveMouse
            - RobotAction
            - RobotActionQueue
            - ScreenShot
          - datepicker
            - DateTimePicker
            - DateTimePickerSkin
            - MainDateTimePicker
          - entity
            - Cryptography
            - Directory
            - FileStream
            - Image
            - Interval
            - LineCounter
            - Session
            - Student
            - StudentView
  - TeacherServer.java
  - ControllerTeacher.java
  - StudentGui.java
  - Threader.java
  - ControllerStudent.java
  - TeacherServer.java
  - LineCounter.java
  - Client.java

```

Code Editor (TeacherServer.java):

```

public static int PORT = 5555;

private final SocketWriterThread writer;
private final SocketReaderThread reader;

public TeacherServer(Socket socket) throws IOException, ClassNotFoundException {
    ObjectOutputStream out = new ObjectOutputStream(socket.getOutputStream());
    ObjectInputStream in = new ObjectInputStream(
        new BufferedInputStream(
            socket.getInputStream()));
    System.out.println("waiting for student name ...");

    LoginPacket packet = (LoginPacket) in.readObject();
    Student student = new Student(packet.getUserName(), packet.getDirOfWatch());
    System.out.println("I got the Package: " + packet.getDirOfWatch());
    Session.getInstance().loginStudent(student);

    reader = new SocketReaderThread(student, in, this);
    writer = new SocketWriterThread(student, out);

    reader.setDaemon(true);
    writer.setDaemon(true);

    writer.handOut();

    reader.start();
    writer.start();

    System.out.println("finished connecting to " + socket);
}

```

Run Log:

```

2: Favorites
Run: TeacherGui (1) StudentGui (1)
  jobs = [ScreenShot(1.0)]
  jobs = [ScreenShot(1.0)]
  jobs = [ScreenShot(1.0)]
  create zip file handInFile.zip ...
  created zip archive
  sending handInFile.zip ...
  sending completed: handInFile.zip
  ERFOLGREICH ==> true
  Connection closed abruptly: java.net.SocketException: Socket closed
  Directory /Users/Philipp/Desktop/Gurca already exists!
  fetching file angabe.zip ...
  fetching completed: angabe.zip
  jobs = [ScreenShot(1.0)]
  jobs = [ScreenShot(1.0)]
  jobs = [ScreenShot(1.0)]

```

Bottom Status Bar:

All files are up-to-date (2 minutes ago)

Teacher Client

Forster
Froschauer
Furtak
Gaisch
Giurca
Gnadlinger
Hermüller
Isla
Klepic
Korthauer
Krannich
Melhorn
Mosquera
Pautz
Pohn
Pühringer
Rudelsdorfer
Saiz
Steininger
Stelzmüller
Tanzer

Optionen Student-Details Abgabe

Lines of Code in Project

● Hermüller/19:46:48.037

Project Structure:

```

TestumgebungNeu
  +-- src
    +-- main
      +-- java
        +-- at
          +-- htl
            +-- remotecontrol
              +-- entity
                +-- Session.java
      +-- LineChartWithHover.java
      +-- HoveredThresholdNode.java
      +-- Directory.java
      +-- Session.java
      +-- Student.java
  +-- TeacherGui.java
  +-- Threader.java
  +-- packets
    +-- HandInPacket.java
    +-- HandOutPacket.java
    +-- LoginPacket.java
    +-- ResponsePacket.java
  +-- server
    +-- SocketReaderThread.java
    +-- SocketWriterThread.java
    +-- TeacherServer.java
  +-- student
    +-- Client.java
    +-- LineChartWithHover.java
  +-- META-INF
    +-- MANIFEST.MF
  +-- resources
    +-- fxml
      +-- CloseRequest.fxml
      +-- Student.fxml
      +-- Teacher.fxml
    +-- images
      +-- checked.png
      +-- falseWay.gif
      +-- inArbeit.gif
      +-- loading.gif
      +-- planen.gif

```

Code Editor (ControllerTeacher.java):

```

Long _minutes = now.until(LocalDateTime.now(), ChronoUnit.MINUTES);
now = now.plusMinutes(_minutes);

Long _seconds = now.until(LocalDateTime.now(), ChronoUnit.SECONDS);

Long _time = _seconds + _minutes * 60 + _hours * 60 * 60;

//punkt im diagramm speichern
//student.addLoc(_loc, _time);

Student toModify = findStudentByName(student.getName());
toModify.addLoc_Time(_loc, _time);

//punkt in diagramm anzeigen
Platform.runLater() -> {
    TextField selected = (TextField) StudentView.getInstance().getLv()
        .getSelectionModel().getSelectedItem();
    if (selected != null) {
        //ist es vom ausgewählten Studenten?
        if (student.getName().equals(selected.getText())) {
            XYChart.Series<Number, Number> actual = getSeries();
            chart.getData().remove(actual);

            XYChart.Data<Number, Number> data = new XYChart.Data<>(_time, _loc);
            data.setNode(
                new HoveredThresholdNode(
                    0,
                    _loc
                )
            );
            actual.getData().add(data);
            chart.getData().add(actual);
        }
    }
}

```

Run:

- TeacherGui (1)
- StudentGui (1)
- LineChartWithHover

```

fetching file angabe.zip ...
fetching completed: angabe.zip
jobs = []
jobs = [ScreenShot(1.0)]

```

Persistence:

Event Log: 1 Event Log

All files are up-to-date (3 minutes ago)

Screenshot Statistik

- 20 Schüler * 3 Stunden**
-
- 12.300 Screenshots**
- 8.49 GB**
- 1 Stunde**
- ~205 Screenshots**
- 141,5 MB für alle Screenshots**
- 782KB pro Screenshot.**

Student-GUI

Student

Username: Giurca

Teacher-IP: localhost

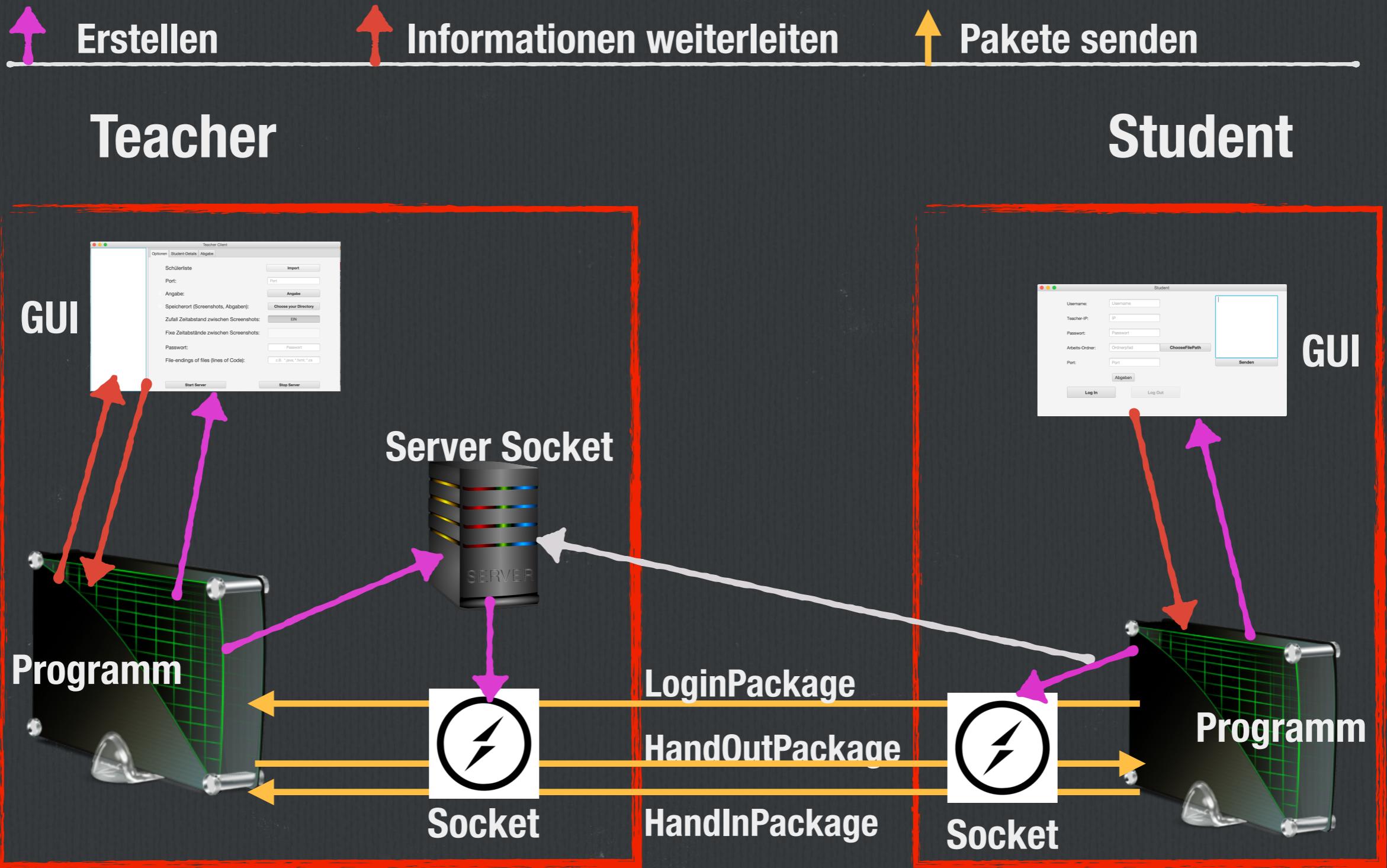
Password: ····

Arbeits-Ordner: /Users/Philipp/Desktop/

Port: 5555

Angemeldet

Umsetzung (Prinzip)



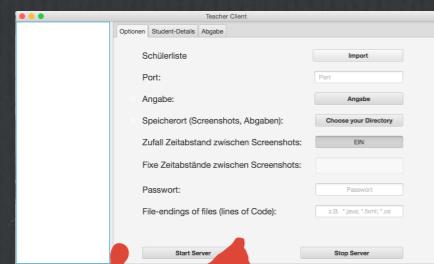
Umsetzung (Harvester)

Teacher

*Screenshot erstellen
und Lines of Code zählen

Student

GUI



Inhalt der Ernte anzeigen

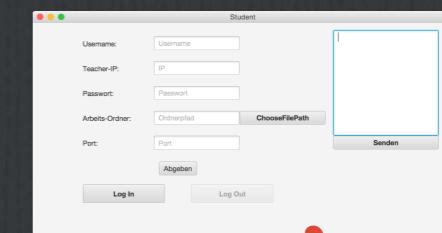
Programm



Socket

sendet Harvester

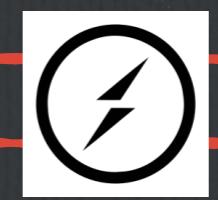
sendet
harvestedPackage



harvested*



Programm



Socket