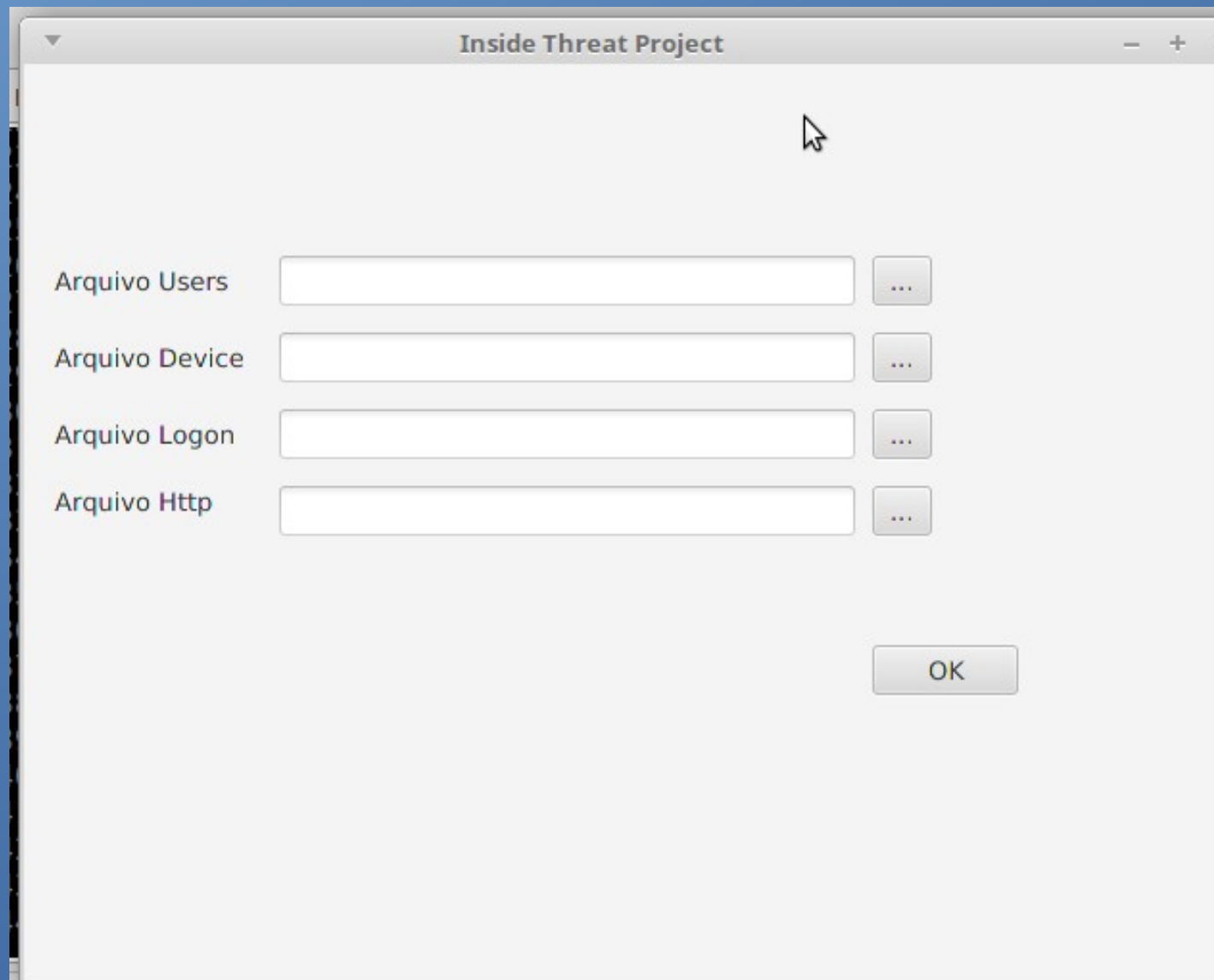


Inside Threat Project

Andrécio Costa Bezerra

Dante Alighieri Miranda dos Santos

Inside Threat Project

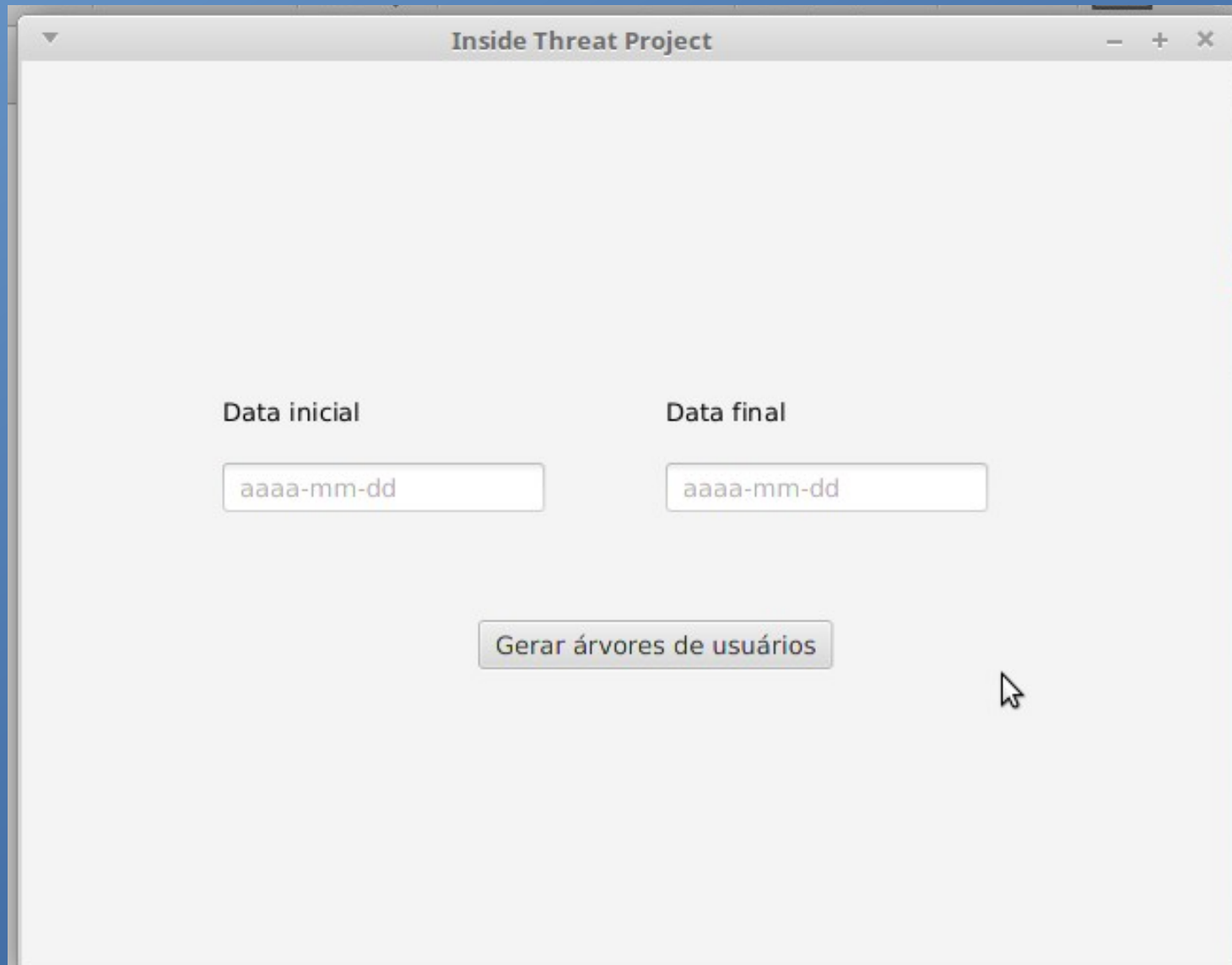


The screenshot shows a window titled "Inside Threat Project" with a standard macOS-style title bar (minimize, maximize, close buttons). The window contains four rows of input fields, each labeled on the left and followed by a text box and a small button with three dots (indicating a file picker):

- Arquivo Users
- Arquivo Device
- Arquivo Logon
- Arquivo Http

At the bottom right of the window is a large "OK" button. A mouse cursor is visible near the top center of the window.

Inside Threat Project



The screenshot shows a web application window titled "Inside Threat Project". The interface is minimalist with a light gray background. It features two input fields for dates, labeled "Data inicial" and "Data final", both containing the placeholder text "aaaa-mm-dd". Below these fields is a single button labeled "Gerar árvores de usuários". A mouse cursor is visible near the bottom right of the button.

Inside Threat Project

Data inicial

aaaa-mm-dd

Data final

aaaa-mm-dd

Gerar árvores de usuários

Inside Threat Project

```
public class BuildForest {  
    String startTime;  
    String endTime;  
    boolean userTree;  
    ArrayList<Node> forest;  
    ArrayList<Device> devices;  
    ArrayList<Http> https;  
    ArrayList<Logon> logons;  
  
    public BuildForest(String start_at, String end_at, boolean ut) {  
        startTime = start_at;  
        endTime = end_at;  
        userTree = ut;  
        forest = new ArrayList<Node>();  
    }  
}
```

Inside Threat Project

```
public ArrayList<Node> run() {  
  
    DeviceDAO dDAO = new DeviceDAO();  
    HttpDAO hDAO = new HttpDAO();  
    LogonDAO lDAO = new LogonDAO();  
  
    try {  
        devices = dDAO.betweenDates(startTime, endTime);  
        https = hDAO.betweenDates(startTime, endTime);  
        logons = lDAO.betweenDates(startTime, endTime);  
    } catch (SQLException e) {  
        System.out.println(e.getMessage());  
    }  
  
    buildUserNodes();  
  
    return forest;  
}
```

Inside Threat Project

```
private void buildUserNodes() {  
    for (Device d : devices)  
        setBuildNode("Device", d);  
    for (Http h : https)  
        setBuildNode("Http", h);  
    for (Logon l : logons)  
        setBuildNode("Logon", l);  
}
```

Inside Threat Project

```
private void setBuildNode(String kind, Activity a) {  
    ArrayList<String> branch = new ArrayList<String>();  
    branch.add(a.getDate());  
    branch.add(a.getPcId());  
    branch.add(kind);  
    branch.add(a.getActivity());  
  
    for (Node n : forest) {  
        if (n.getKey() == a.getUserId()) {  
            n.addNodeInChain(branch);  
  
            return;  
        }  
    }  
  
    Node node = new Node(a.getUserId());  
    node.addNodeInChain(branch);  
    forest.add(node);  
}
```

Inside Threat Project

```
void setBuildForest(ActionEvent event) {
    BuildForest b = new BuildForest(startDate.getText(), endDate.getText(), false);
    ArrayList<Node> forest = b.run();
    System.out.println(forest.size());
    try {

        FXMLLoader loader = new FXMLLoader();
        loader.setLocation(MainScreenController.class.getResource("../views/ChooseUser.fxml"));
        AnchorPane chooseUserScreen = (AnchorPane) loader.load();
        ChooseUserController controle = new ChooseUserController();
        controle.carregarUsuarios(forest);

        mainScreen.setCenter(chooseUserScreen);
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```