

Software Development

Guided Exercise 3

David Atwood - 100480182
Kater Ennada Helali - 100481030

FUNCTIONALITY 1

Evaluation of Group 15's test cases

- (+) Description successfully explains the nature of each test case.
 - (+) Invalid input is highlighted in red which makes it very easy to distinguish the error handled.
 - (+) The rest of the input is consistently held constant in each test case.
 - (+) I/O correctly displayed.
 - (+) Thorough test cases analyzing all possible downfalls.

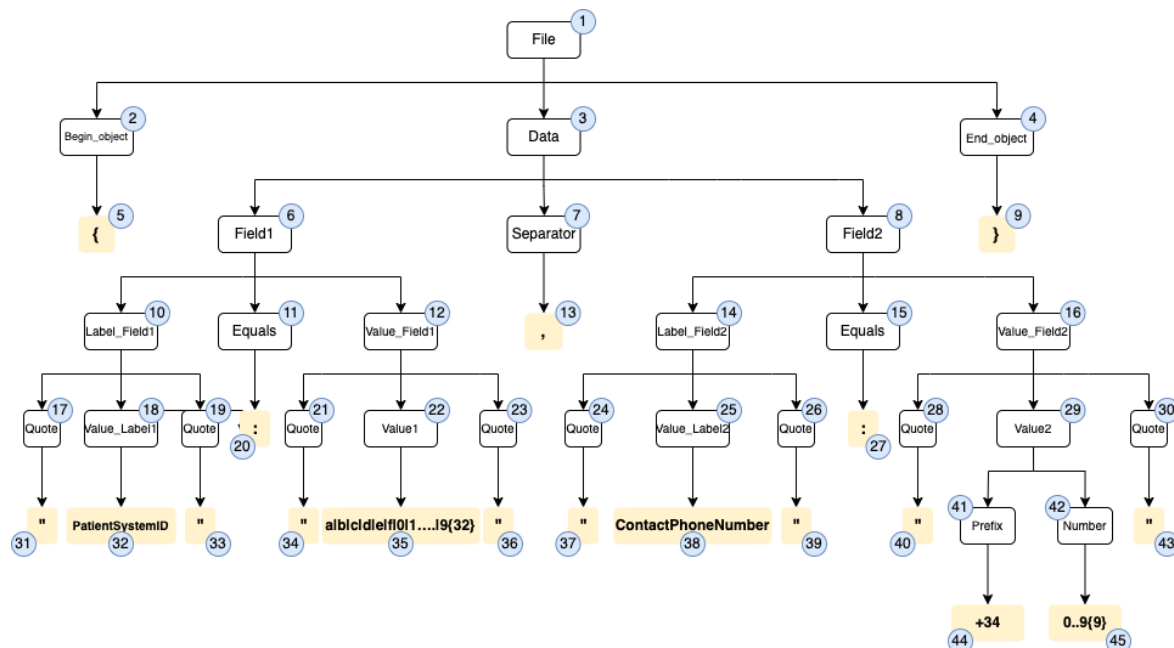
 - (-) Four ID tests grouped together in the first row for no clear reason.
 - (-) The criteria column is a replica of the description. The statement criteria that each case is testing, therefore, missing.
- => Overall good report.

FUNCTIONALITY 2

Definition of grammar:

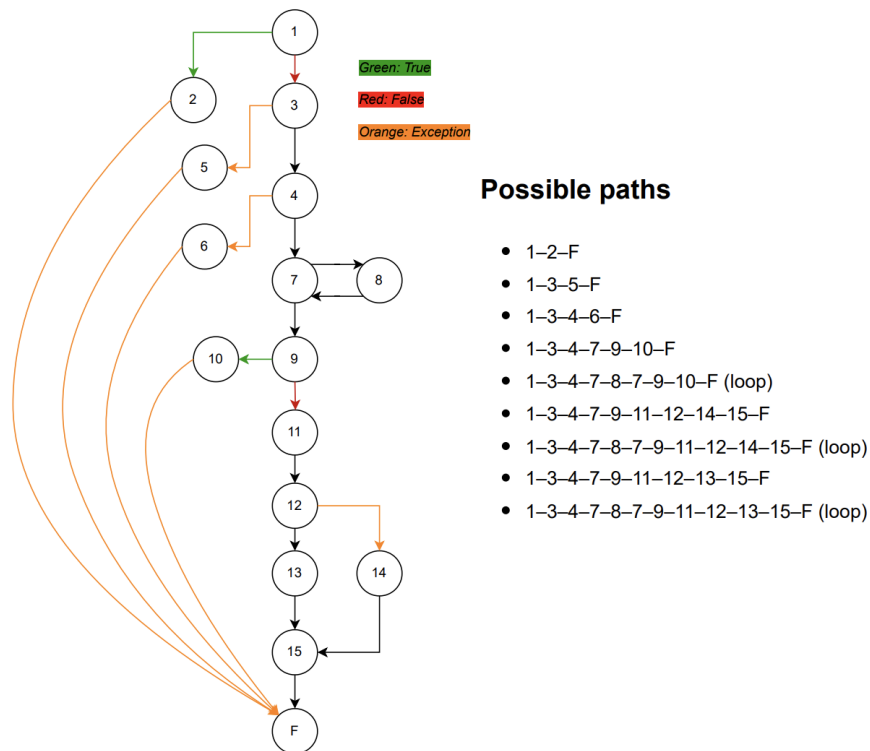
- **File**:= Begin_object, Data, End_object
- **Begin_object**:= {
- **End_object**:= }
- **Data**:= Field1, Separator, Field2
- **Field1**:= Label_Field1, Equals, Value_Field1
- **Separator**:= ,
- **Field2**:= Label_Field2, Equals, Value_Field2
- **Equals**:= :
- **Label_Field1**:= Quote, Value_Label1, Quote
- **Quote**:= "
- **Value_Label1**:= PatientSystemID
- **Value_Field1**:= Quote, Value1, Quote
- **Value1**:= a|b|c|d|e|f|0|1...|9 {32}
- **Label_Field2**:= Quote, Value_Label2, Quote
- **Value_Label2**:= ContactPhoneNumber
- **Value_Field2**:= Quote, Value2, Quote
- **Value2**:= Prefix, Number
- **Prefix**:= +34
- **Number**:= 0..9 {9}

Derivation tree



FUNCTIONALITY 3

Control flow graph



Corresponding code

```
def vaccine_patient(self, date_signature):
    if not re.search('[0-9a-fA-F]{64}', date_signature):
        raise VaccineManagementException("date_signature is invalid")

    try:
        with open(self.vaccination_appointments, "r", encoding="utf-8") as file:
            appointments = json.load(file)
    except FileNotFoundError as ex:
        raise VaccineManagementException("vaccination_appointments file not found") from ex
    except json.JSONDecodeError as ex:
        raise VaccineManagementException("vaccination_appointments file is not in valid JSON format") from ex

    match = False
    for appointment in appointments:
        if date_signature == appointment["vaccinationAppointment_vaccination_signature"]:
            match = True

    if not match:
        raise VaccineManagementException("date_signature not found in vaccination_appointments")

    now = datetime.utcnow()
    new_administration = {"VaccinationSignature": date_signature, "Timestamp": datetime.timestamp(now)}
    # add administration to file, creating the file first if necessary
    try:
        # if file does not exist store the first item
        with open(self.vaccination_administration, "x", encoding="utf-8", newline="") as file:
            data = [self._dict_]
            json.dump(data, file, indent=2)
    except FileExistsError:
        # if file exists, load the data, append the new item and save it all
        with open(self.vaccination_administration, "r", encoding="utf-8") as file:
            # LOAD THE DATA
            data = json.load(file)
            # APPEND THE NEW ADMINISTRATION
            data.append(new_administration)
        # Overwrite the data
        with open(self.vaccination_administration, "w", encoding="utf-8", newline="") as file:
            json.dump(data, file, indent=2)

    return True if match else False
```