

北京邮电大学

# 用户使用手册

DSL-程序设计实践

[作者姓名]陈朴炎

学号：2021211138

2023-11-29

## 目录

1 依赖环境.....	2
2 运行项目/使用手册.....	2
3 记法.....	4
3.1 定义初始状态.....	4
3.2 定义状态、回应及状态转移.....	5
3.3 定义默认的应答逻辑 .....	6
3.4 使用变量.....	7
4 脚本语言的形式化定义 .....	7

## 1 依赖环境

后端环境: python >= 3.10

前端环境: Qt >= 5.12, pyqt

如果您对操作系统环境尚不满足, 请到相关网站寻求帮助:

[python 官方网站](#)

[pyqt 依赖包下载](#)

pyqt 命令行下载:

```
pip install PyQt5
```

在命令行中, 输入 `pip install PyQt5`, 即可安装前端所需的依赖包。

## 2 运行项目/使用手册

1、首先, 要打开服务端, 监听 socket。

打开命令行, 将您的目录切换到 `/DSL/backend`, 然后输入:

```
python3 server.py
```

运行服务器程序, 这时候服务器就将监听本机 ip 地址的 8888 端口。

2、然后, 需要打开客户端, 向服务器发出请求。

打开命令行, 将您的目录切换到 `/DSL/pyqt` 或者是 `/DSL/front_page`, 输入:

```
python3 login_window.py
```

运行前端登陆界面程序, 将会弹出登录界面窗口:

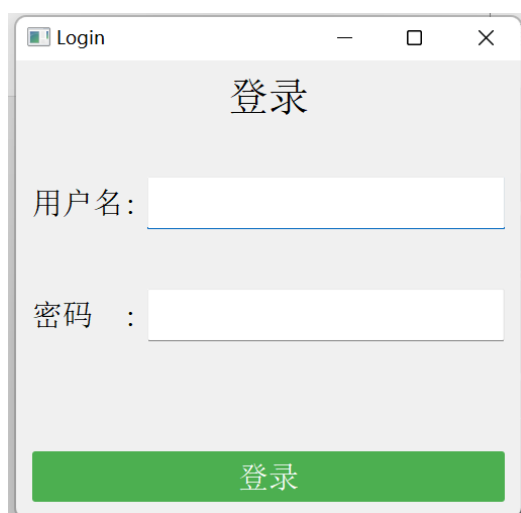


图 2-1 登陆界面窗口示意图

### 3、登录账号

登录界面需要您输入您的用户名和密码，之后点击登录按钮或是按下回车，就可以登录，跳转到聊天界面。

请您输入正确的用户名和密码，否则就会出现一下弹窗：

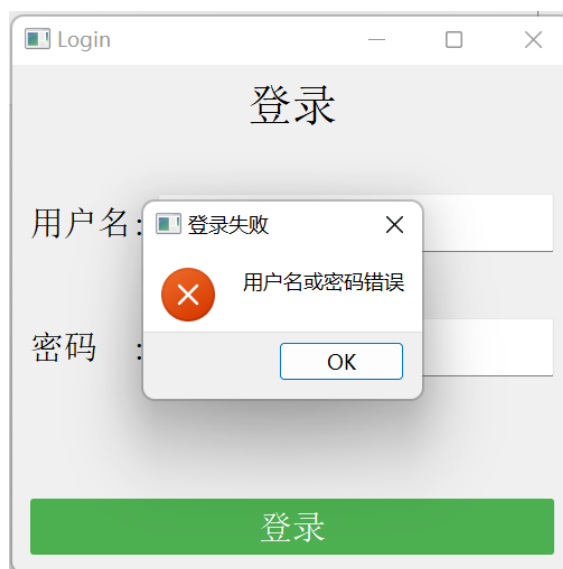


图 2-2 登陆失败示意图

### 4、如何使用会话窗口

当您登录成功后，会跳转到聊天会话页面，客服机器人将会提示您聊天

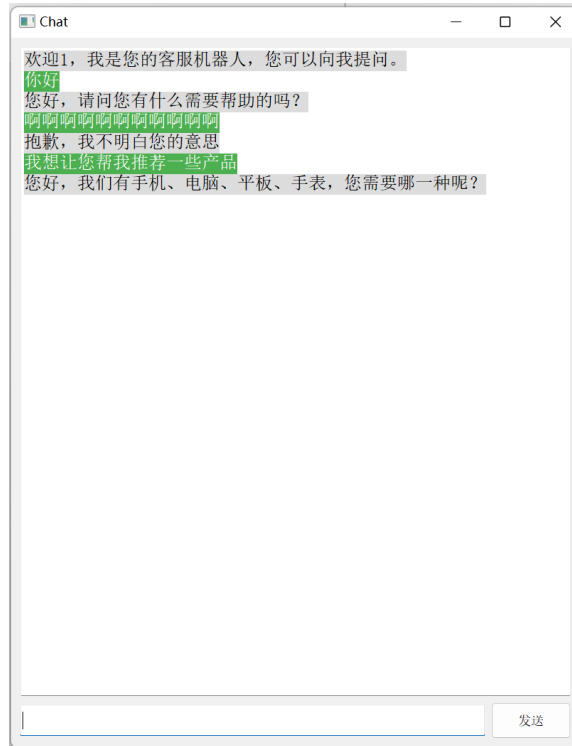


图 2-3 客服会话示意图

如图 2-3 所示, 聊天界面有一个输入栏, 有一个发送按钮, 您需要在输入栏输入您想咨询的信息, 之后点击发送或是按下回车, 便可向机器人询问。您的消息将被展示为白字绿底, 客服的消息将被展示为黑字灰底。

## 3 记法

在线客服机器人的应答逻辑被抽象为一个摩尔型自动机。DSL 脚本即为描述这个自动机的脚本, 它是由对一系列状态的描述组成的。当用户登录后, 将会有有一个初始状态, 下面我们先来看如何定义一个初始状态

### 3.1 定义初始状态

语法: entry "state_name"
示例: entry "初始"

上述代码定义了一个名字叫做 初始 的状态，代表用户登录后将会首先进入到初始状态中。语法是在一行中，entry 后跟空格，再跟双引号，双引号中是初始状态的状态名称。

### 3.2 定义状态、回应及状态转移

语法：

```
state "state_name"
    response "message"
    when "trigger1" state_transfer "state1" response
"message1"
    when "trigger2" state_transfer "state2" response
"message2"
    when "trigger3" state_transfer "state3"
    when "trigger4" response "message1"
    wait 120
```

示例：

```
state "初始"

    response "您好，请问您有什么需要帮助的吗？"

    when "你好|您好" response "您好，请问您有什么需要帮助的吗？"

    when "机器人" response "您好，我是您的客服机器人，有什么需要帮
助的吗"

    when "人工" response "请求人工客服，请拨打 2021211138"

    when "售后|问题" state_transfer "售后" response "现在给您转
售后..."

    when "售前|推荐" state_transfer "售前"
```

- 首先是 state 创建语句，在定义一个状态之前需要用 state “state\_name”

在脚本中声明这个状态，并且，在这句声明和下一个状态声明语句（或文件结束）之间的所有语句都隶属于该状态。

- response 语句：response 语句用来声明该状态的初始回应。在该状态下，可以使用 response 语句来声明跳转到该状态的初始回应。
- when 语句：when 语句用来声明当遇到某个关键词时，将执行什么结果。在该语句中，when 后跟一个触发的关键词，用双引号包裹，再跟上某个动作类型以及动作结果，可以叠加多个动作，表示遇到该关键词将会状态转移或是回应某句话。比如 when "售后|问题" state\_transfer "售后" response "现在给您转售后..." 这句脚本语言就表示，在该状态下，遇到关键词“售后|问题”将会跳转到售后状态，并告诉用户现在给您转售后...
- wait 语句：wait 后跟一个整数，代表超时时间，表示当用户多久没有应答时，将退出。

### 3.3 定义默认的应答逻辑

语法：

```
when    "trigger"    <act_type1    "act_result1">    <act_type2  
"act_reslut2"> <...>  
.....
```

示例：

```
when "你好" state_transfer "初始"
```

```
when "售前|推荐" state_transfer "售前"
```

```
when "售后|问题" state_transfer "售后" response "现在给您转售  
后..."
```

```
when "谢谢|再见|好的|好吧" response "很高兴能帮助到您，祝您生活愉快~"
```

- 请注意：默认的应答逻辑脚本语句不能跟在其他的 state 声明之后，它必须放在脚本文件的开头，或者跟在名字叫做“默认”状态之后
- 其他的用法与普通状态声明并无二意

### 3.4 使用变量

语法：

```
when "trigger" response "$username ..... $order ....."
```

示例：

```
when "订单" response "$username 的订单是: $order"
```

- 想要根据不同的用户返回不同的回应，需要在回应中加入以\$开头的变量。
- 目前为止，变量名只有两类，一个是 username，一个是 order。你可以放在任何 response 中，可以只使用单个的\$username 或单个的\$order 或两者混合使用，服务器将解析解释器返回的语句，并将 username 和 order 替换为用户的名称以及他们的订单信息。

## 4 脚本语言的形式化定义

$S' \rightarrow \text{entry } \langle \text{state\_name} \rangle D$

$D \rightarrow \text{when } \langle \text{trigger} \rangle \langle \text{act\_list} \rangle D \mid S$

$S \rightarrow \text{state } \langle \text{state\_name} \rangle F$

$F \rightarrow \text{response } \langle \text{response} \rangle F$

$F \rightarrow \text{when } \langle \text{trigger} \rangle \langle \text{act\_list} \rangle F \mid S \mid \epsilon$

$\langle \text{act\_list} \rangle \rightarrow \text{response } \langle \text{response} \rangle \langle \text{act\_list} \rangle \mid \text{state\_transfer } \langle \text{state\_name} \rangle \langle \text{act\_list} \rangle \mid \epsilon$

$\langle \text{state\_name} \rangle \rightarrow \text{"\%一个状态名称\%"}$



<response> -> “%一句回应%”

<trigger> -> “%某个触发关键词%”

其中，%%百分号包裹起来的代表任意字符串。