# 《心·晴心理咨询系统》项目部署

**第九组**

**v1.0**

# 一、前端部署

## 1.1 uniapp配置文件

在UniApp项目中，manifest.json文件用于配置应用的基本信息、发行平台参数、权限声明、网络配置和页面路径等内容，确保应用在不同平台上正确运行。

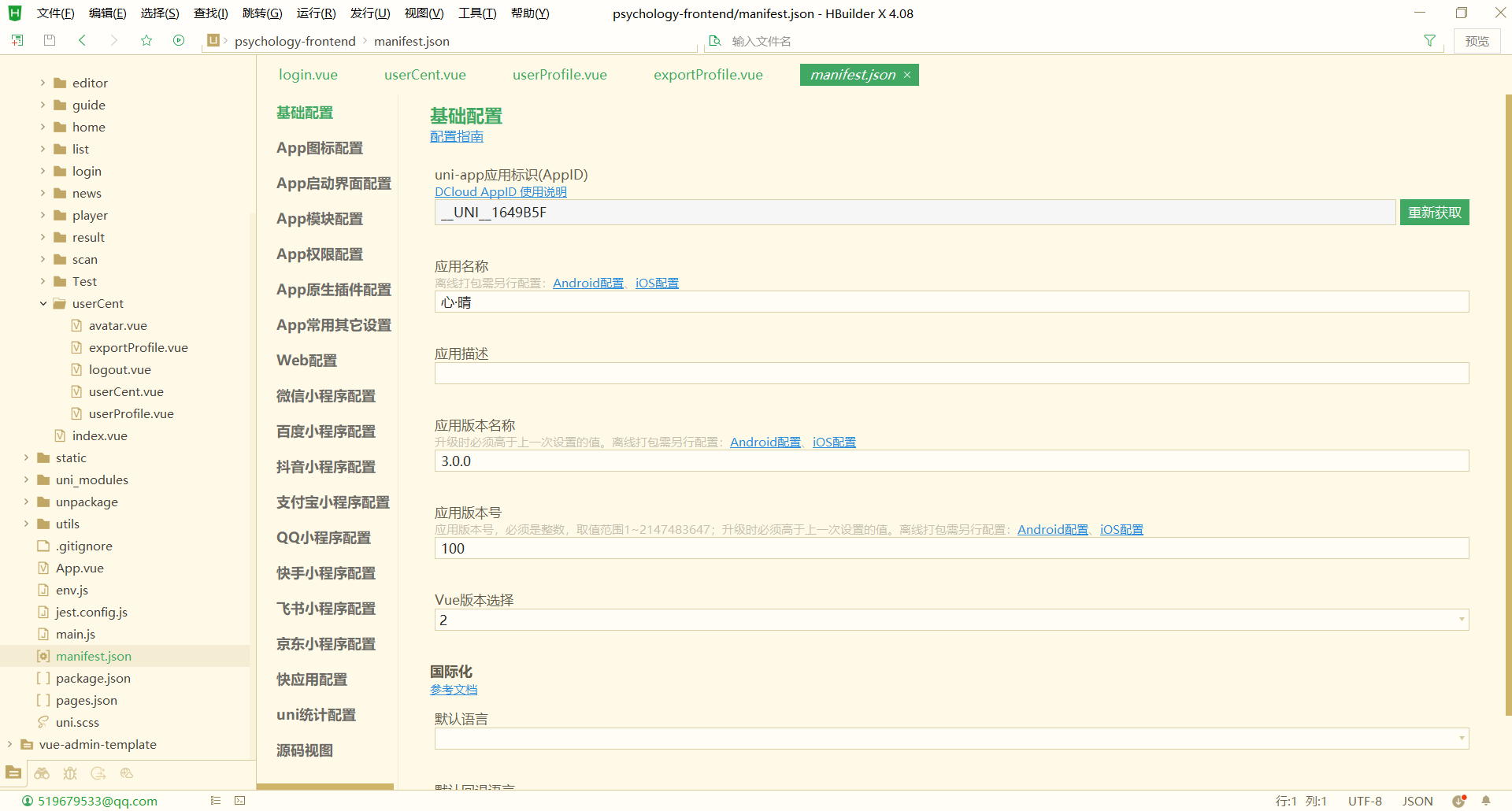


图1.1 manifest.json配置文件

## 1.2项目发行

### 1.2.1 云打包

使用云端服务来打包生成原生应用程序，编写跨平台代码后，通过云打包服务自动生成原生应用程序，从而省去本地环境配置和复杂的打包步骤，使得打包过程更加便捷和高效。

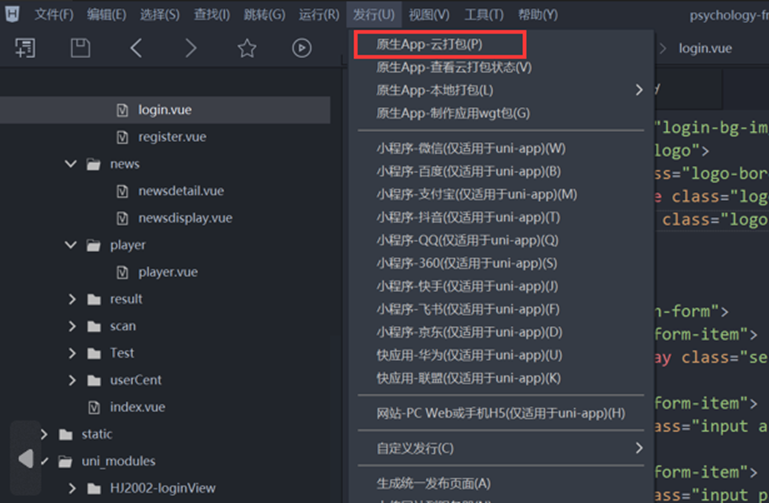


图1.2.1 原生app云打包服务

### 1.2.2填入打包配置信息

在进行原生app云打包时，需要填写应用的基本信息（名称、版本号、描述等）、AppID和包名、图标和启动图、权限声明、证书信息（如iOS的开发者证书和Android的签名密钥信息）以及平台特定配置（如iOS的Apple ID和Android的应用市场信息），确保生成的原生应用符合各平台要求并能正常发布和运行。



图1.2.2 打包配置信息

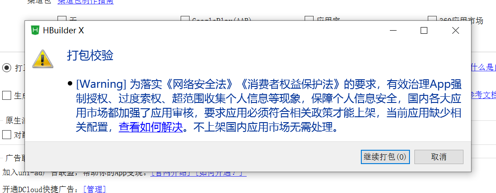


图1.2.3 打包校验

### 1.2.3打包成功

HBuilderX中的云打包过程包括多个步骤：首先，上传项目并选择使用Android云端证书，然后云端开始打包，这个过程通常需要2-5分钟，接着在本地生成安装包并存储在指定路径，完成后可以选择进行应用安全加固或将安装包上传到UniCloud进行管理。



图1.2.4 云打包成功

# 二、后端项目部署

为了减少手动输入命令的繁琐并简化操作流程，我们将启动步骤全部集成到一个脚本中，以确保每次执行的命令都是一致和正确的。使用dockercompose一键部署，配合restart.sh脚本中的启动命令，启动时只需输入sudo sh ./restart.sh 即可启动。

restart.sh脚本使用Docker和Maven进行后端项目的构建和部署，具体步骤如下：

①关闭当前运行的Docker容器，并删除本地镜像。

②进入后端项目目录./backend。

③执行mvn install命令，安装项目所需的依赖包。

④如果安装成功，接着执行mvn package命令，打包项目。

⑤返回到原来的目录。运行docker-compose up -d命令，启动Docker容器。

脚本中包含了错误检查机制，在每个关键步骤失败时会输出错误信息并退出脚本执行。

restart.sh文件内容如下：

#!/bin/sh

sudo docker-compose down --rmi local

# Navigate to ./backend directory

cd ./backend || { echo "Failed to change directory to ./backend"; exit 1; }

# Execute mvn install

echo "Executing mvn install in ./backend..."

mvn install

**if** [ $? -ne 0 ]; **then**

echo "mvn install failed."

exit 1

**fi**

# Execute mvn package

echo "Executing mvn package in ./backend..."

mvn package

**if** [ $? -ne 0 ]; **then**

echo "mvn package failed."

exit 1

**fi**

# Navigate back to the original directory

cd - || { echo "Failed to change back to the original directory"; exit 1; }

# Execute docker-compose up -d

echo "Executing sudo docker-compose up -d..."

sudo docker-compose up -d

**if** [ $? -ne 0 ]; **then**

echo "docker-compose up failed."

exit 1

**fi**

echo "Script executed successfully."

脚本包含运行成功的提示信息，效果如下：

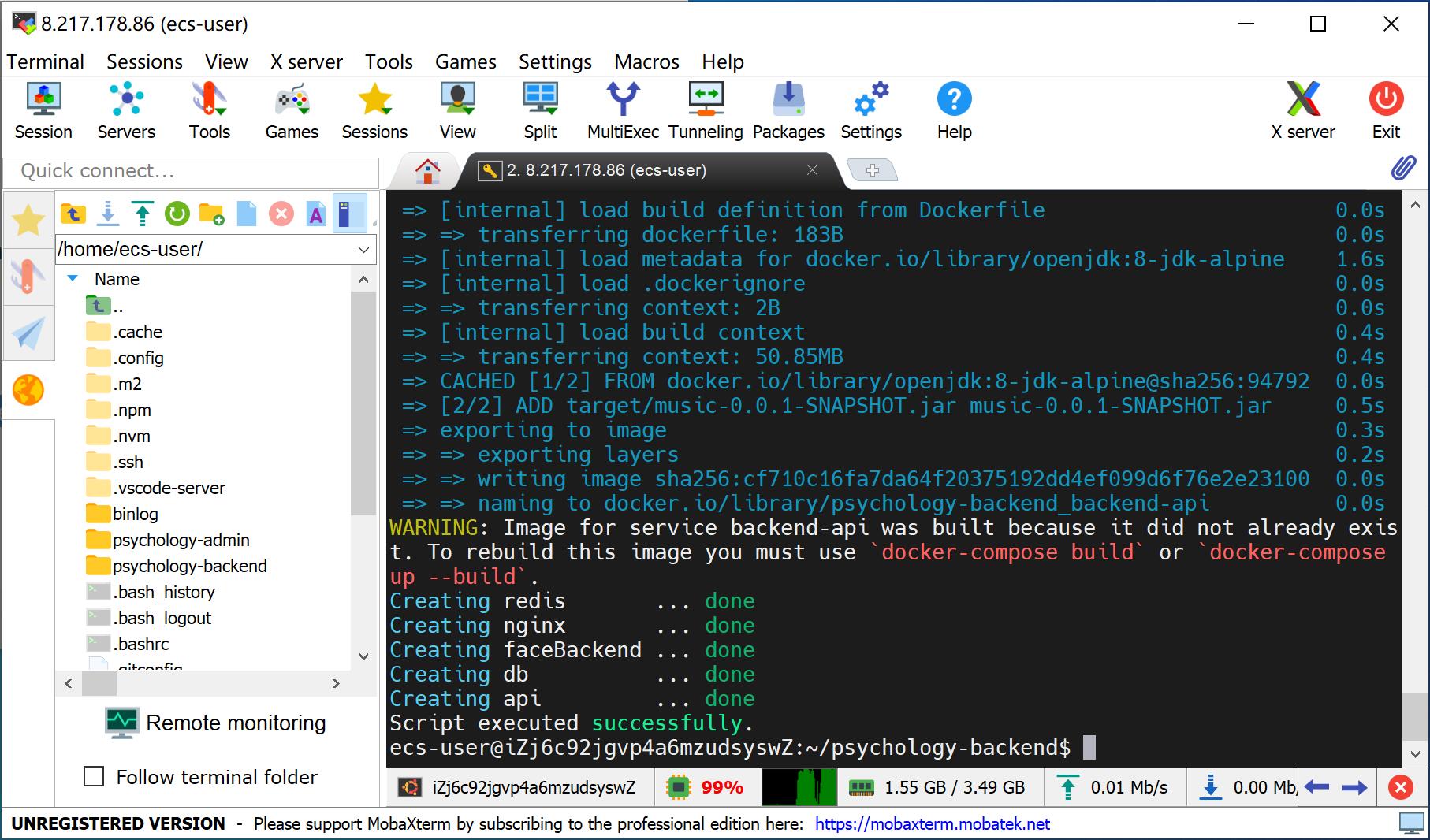


图2.1 脚本执行成功