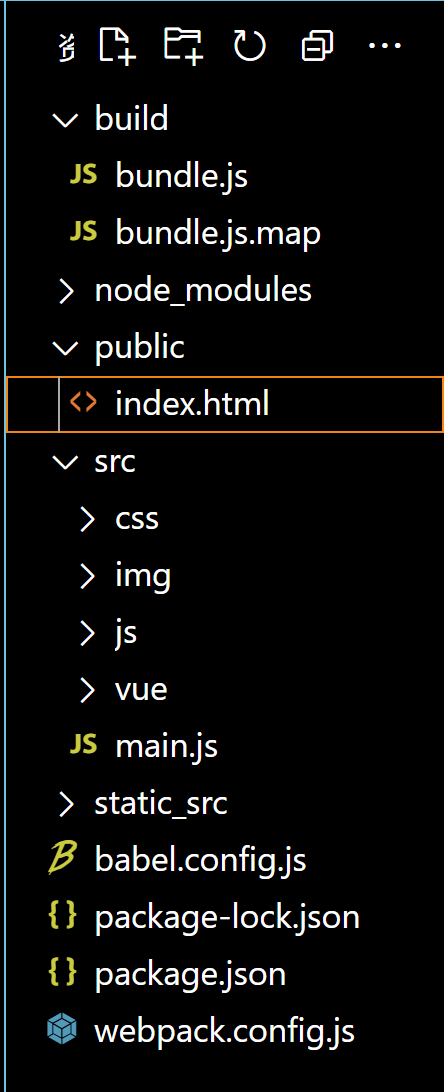
原型进展

前端原型

在本项目中，我们使用VUE3框架开发前端，从而进行HW的网页设计。同时，我们还使用了npm作为前端项目包管理工具，并使用webpack进行打包。

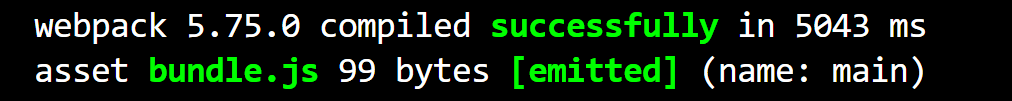
以登录界面为例

文件目录如下：

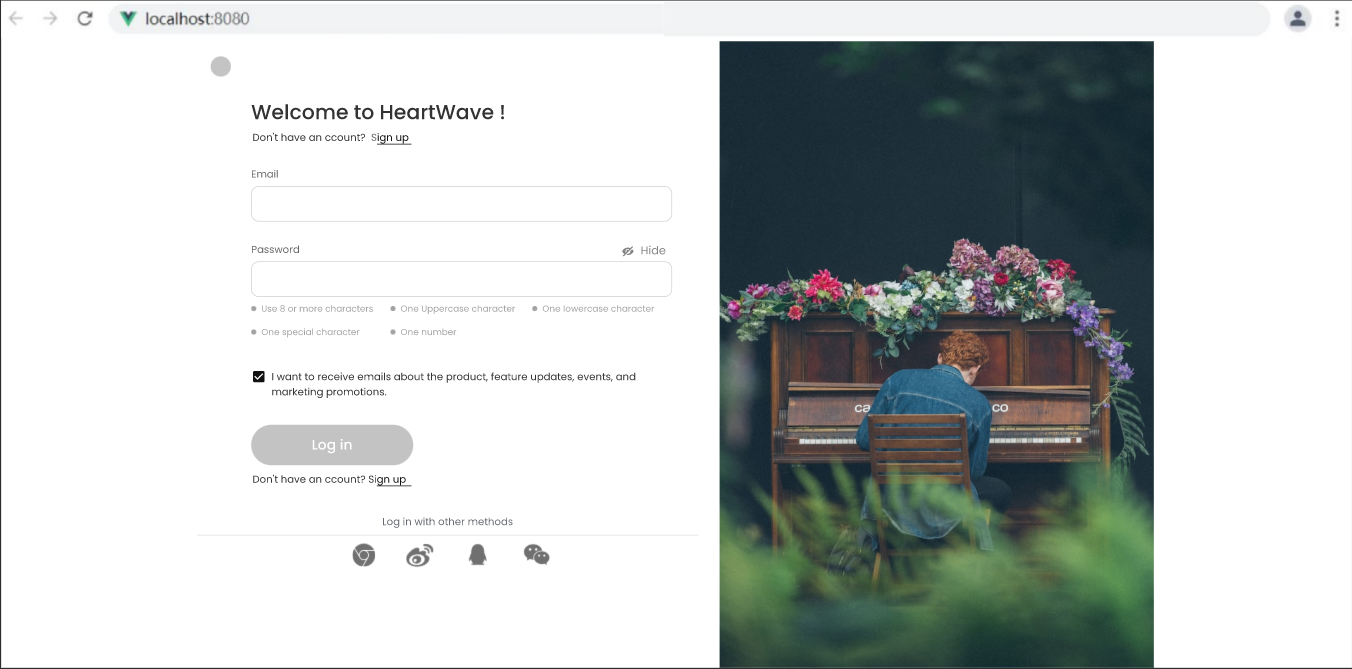


我们在终端中输入`npm run serve `指令，便可以在localhost为8080的端口访问到我们的页面。





我们的登录页面如下：



后端原型

在本项目中，我们使用Java进行后端的开发。为了将前后端连接起来，我们使用相应的API保证两个部分的通信。

以下，我们将以获得用户情绪数据为例进行展示。该请求为GET请求，REST API为/api/emotion-analysis/history，参数为请求返回心情历史记录的用户的id。

首先，我们需要使用Java语言建立一个相关用户情绪数据的类以及存储该类和通过id寻找到某个实例的方法，如下所示：

@Data

@Entity

public class UserMoodData {

    @Id

    private String userid;

    private Date date;

    private int moodcount;

    private Map<String, Integer> mood;

}

@Repository

public interface UserMoodDataRepository extends JpaRepository<UserMoodData, String> {

    UserMoodData findOneById(String id);

    Iterable<UserMoodData> mostRecent(int t):

}

因此，我们可以确定该功能相应的API函数

首先，我们初步定义一些常见的HTTP状态码，帮助我们更好地获得API请求结果的状态。

如下所示：

public enum HttpStatus {

    OK(200, "OK"),

    CREATED(201, "Created"),

    ACCEPTED(202, "Accepted"),

    NO\_CONTENT(204, "No Content"),

    BAD\_REQUEST(400, "Bad Request"),

    UNAUTHORIZED(401, "Unauthorized"),

    FORBIDDEN(403, "Forbidden"),

    NOT\_FOUND(404, "Not Found"),

    METHOD\_NOT\_ALLOWED(405, "Method Not Allowed"),

    INTERNAL\_SERVER\_ERROR(500, "Internal Server Error"),

    SERVICE\_UNAVAILABLE(503, "Service Unavailable");

}

然后，我们通过用户id获得情绪数据的API函数如下：

@RestController

@RequestMapping("/api/emotion-analysis")

public class UserMoodDataController {

    @Autowired

private UserMoodDataRepository userMoodDataRepository;

    @GetMapping("/history/{id}")

    public ResponseEntity<?> getUserMoodDataById(@PathVariable String userid) {

        try {

            UserMoodData moodData = userMoodDataRepository.findOneById(userid);

            if (moodData != null) {

                return new ResponseEntity<>(moodData, HttpStatus.OK);

            } else {

                return new ResponseEntity<>("User mood data not found", HttpStatus.NOT\_FOUND);

            }

        }

        catch (Exception e) {

            return new ResponseEntity<>("Error retrieving emotion data", HttpStatus.INTERNAL\_SERVER\_ERROR);

        }

    }

}

在Apifox中测试如下：

