1.明确功能，我们的项目应该属于偏实用类的项目，与算法类项目不同，我觉得如果我们能取得一个理想的成绩，实用类的项目一定要把功能做全。我们的项目可能一开始的构思偏硬件一些，不过要想成为一个成型的项目还是要有完整的软件和完整的功能实现。

比如共享电池要帮助用户寻找电池，监测电池状况以外。其后面的配套功能，比如信誉积分的评定和支付方式的问题，甚至是电池报警后的售后问题和救援功能都是值得进一步思考的。此外，目前国内充电桩的建设也是日益加速，我们项目也可以提示用户哪里有充电桩以及充电桩的使用情况等等，并且以可视化的形式展现出来；甚至可以加入电池问题反映的功能或者是新能源电池车主沟通的论坛版块，为他们提供一站式的服务，“垄断”新能源电池行业。

这就需要，我们需要设计一些数据库的表格，先想好我们需要哪些数据，构造ER模型以后建表，优化范式，然后进行数据库的操作。

虽然目前我的硬件还需要一段时间，但是大家可以思考更多的功能和他们的实现方式，其实硬件只是我们项目的一部分（我觉得不超过30%），真正重要的功能在于服务和给用户的友好体验，我们既然是实用类项目就应该做的全面一些，没有算法的复杂那就必然要构造复杂的功能和复杂的数据库。

所以我觉得，

第一阶段：1.1调通硬件，解决共享电池帮助用户寻找电池，监测电池状况。1.2开发后面的配套功能，比如信誉积分的评定和支付方式的问题，甚至是电池报警后的售后问题和救援功能问题。

第二阶段：2.1从充电站的角度考虑，提示用户哪里有充电桩以及充电桩的使用情况等等，并且以可视化的形式展现出来；2.2甚至可以加入电池问题反映的功能或者是新能源电池车主沟通的论坛版块，为他们提供一站式的服务。

（有点不确定如果功能太多小程序会不会带不动）

2.技术实现：大家可能觉得我们项目实现起来很困难，其实不然。我这学期的程序设计实验就是利用c++的ODBC界面和微信小程序做一个日程备忘提醒功能，虽然功能比较简单，但是也算是打通了小程序-php-mysql-阿里云数据库-阿里云服务器的过程。我觉得php是后端实现的很好方式，（当然小程序也可以是node.js实现，这个看大家的掌握情况了，如果大家之前都掌握了node.js我们可以转到node.js中，目前我个人的情况是对php比较熟悉）php很好的支持了mysql数据库，其中的php\_mysqli模块就是可以很方便的和前端和后端数据库进行交互，起到了桥梁的作用。再说说后端的mysql数据库，目前最亟待解决的就是建表问题，硬件这边的变量参数我后续会总结出来，与此同时大家也可以想想别的功能需要哪些变量和表结构（兴睿和天秋可能要更辛苦一些）。后续会直接用云端的数据库和服务器跑程序了，这样可以省去本地调试，再部署到云端的过程。（当然也可以先在本地开发，如果经费到位了可以开几个数据库，这样我们可以分开调试和开发）。

总结下来的技术点：(纯属是我个人的理解，毕竟我对很多知识才刚刚起步，我觉得这样的架构是可以实现的，希望大佬能提出更好的建议)

1. **功能规划，一定先把功能想清楚**；
2. 小程序的开发和语法（建议大家系统的看一个教程/网课或者买本书，我618买了小程序，arduino，php有关的书，感觉比网上的资料好太多了）；微信小程序到最后其实就是个前端展示的工具而已，需要一直从后端拿数据。（当然用户的输入指令也需要写入数据库）
3. Php:php\_mysqli（可以重点学这一部分）；
4. 数据库需要的变量，ER图，表结构的优化（这部分建议学相关的课，逻辑性比较强，我选了信通的数据库这部分也是重难点），SQL编程（通常嵌入在php中，不是很复杂，所以一定先把表建好，表有了SQL编程不是问题）；
5. 硬件端和数据库的互联以及硬件的实现，如果硬件可以把数据传输到数据库上，那么小程序的调用也不是问题了。（硬件端最近找了一些比较对接的资料，希望短时间内可以完成）
6. 产品外形和产品前端界面和交互方式的设计。
7. 报告。

3.技术wiki

由于大创项目周期长，而且涉及到开学以后的各种琐事，可能我们不能连续的开发项目，而且技术点不止涉及到一次，如果隔一段时间很容易忘了接口怎么造的或者是之前学习的语法和功能遗忘。之前雏燕计划，就拿我自己举例子，可能寒假学习的小程序和硬件的知识现在都有点忘了，一些资源也在收藏夹里吃灰，甚至找不到了，于是我想了一下，能不能用这种技术wiki的方式。比如我学习到小程序与数据库连接的时候，我可以写一下具体实现的代码，用到哪些资源，参考了哪些帖子，哪些书；做硬件GPS模块的时候，有哪些参数，arduino代码里面需要注意哪些，尤其是一些程序接口特别重要，比如GPS给数据库返回什么数据（经纬度，海拔等等）对应数据库中的什么字段，在小程序中怎么调用，都要写清楚，方便别的同学去调用。

我觉得写这种wiki确实对大家来讲是一种压力，我自己也不知道能不能坚持下去，但是如果我们五个人学会了比较重要的知识都分享出来，整合出来，对彼此也是一种促进，团队的沟通也不容易“死机”，不容易出现沟通断层的情况。总结来说，我个人觉得技术wiki整体上有两点好处：1.整合知识，方便日后查看，也是对自己的一种提升。2.增加团队协作沟通，相互促进。 另外Wiki中的内容，大家可以只写自己认为关键的技术实现，但是涉及到程序接口的一定要写一个详细一点的wiki，

目前想的是在github上创建项目，然后git有代码管理和版本回滚功能，而且小程序的代码也可以直接上传到git上（微信开发者工具自带的版本管理功能默认是传到github上），而且github也支持团队wiki的撰写。但github缺点也在于全英文，可能需要一定的学习成本和访问时间的延迟。（不过我认为还好，而且github我相信大家也都接触过）如果大家认为github不太好的话，我们可以在qq群里面分享想法。

4.讨论分工和需要学习的技术。