Ps：CommunityUtil是自己写的类拥有生成随机字符串的方法（封装了uuid）以及MD5加密方法

可以将用户注册功能分为三个请求，访问注册页面，提交注册数据，激活注册账号（按请求来开发用户注册功能）

访问注册页面：在controller层编写响应方法，返回注册页面路径

提交注册数据：通过maven导入commons lang的jar包，为了判断常用的数据是否空值，在*UserService*层中编写一个注册方法，传入的参数类型为user，返回值为map因为可能是某一字段不能为空，账号已存在等等信息，首先对空值处理，判断user是否为null，是的话抛出异常，在判断user里的属性是否为空，若为空将错误信息put（key=“usernamemsg”，value=“账号不能为空” 等映射）进返回的map中，直接ruturn map，若上述错误都没有判断username、Email是否重复，若重复 return，都没有问题，将用户设置的密码进行加密（通过CommunityUtil设置salt与用户密码拼接在一起之后传入给CommunityUtil的加密方法）覆盖用户密码，将一些用户没有编写的属性赋值（type、status等），赋给用户一个随机头像（应该传入路径参数，共有1000个随机头像，大体的路径相同，只有一个数字不同可以通过random随机生成），用insertUser方法导入数据库中，给用户发邮件方法（让其激活），将context（对应email以及对应的网址（域名+项目访问路径+项目下激活的访问功能路径+用户id+激活码））和网站模板传给模板引擎，调用发送邮件方法传入用户邮箱，主题为激活账号，内容为模板引擎生成的网站，在*LoginController*层的注册方法中首先得到userservice的注册方法返回的map，如果map为空说明成功注册，返回一个说明注册已成功，且过几秒会跳转到首页的网站模板路径（其中要将动态值（mas：注册成功，我们已经向您的邮箱发送了一封邮件，请尽快激活；跳转的网页目标：首页路径）加入进模板），如果失败在model中分别添加用户名，密码，邮件等的错误信息return注册网站的路径，其中当返回注册页面时，想要保存上次的填写信息（错误的信息），可以在模板中设置一个判断，比如返回上次的用户名填写信息，先判断user是否为空，若不为空则填入nsername，错误时有is-invalid标志

激活注册账号：不用编写dao层，已经在数据库中有了，只是没有激活，在UserService层：编写激活方法（参数为userId，激活码code），通过userId查到user对象，判断user对象的状态（status）若为1则说明已经激活过，返回2为重复激活，如果激活码与传入激活码一样则先将用户状态改为1，返回0成功激活，否则（激活码不对）返回1激活失败，在*LoginController*层处理请求，路径为网址，方法参数为model，userId（通过@PathVariable得到路径中对应的userId值（被{}括住，且名字相同），code值（激活码，获取方法同上）），返回的是跳转模板的路径，调用service的激活方法赋给result，判断result的值，若激活成功，要将动态值（mas：激活成功，您的账号已经可以正常使用；跳转的网页目标：登陆页面路径）加入进model，若重复激活，将动态值mas：无效操作，该账号已经激活过了；跳转的网页目标：登陆页面路径加入进model，若激活失败，将动态值（mas：激活失败，您提供的激活码不正确；跳转的网页目标：首页路径）加入进model