

# Bullet弹幕时间胶囊

## 项目提案

大数据背景下很多所谓的线上社交活动其实没有什么现实意义，即，这些活动极大可能不会对我们的生活轨迹产生积极影响；相反的，它们还可能给我们的生活带来不要的困扰。

以Bilibili弹幕视频网站为例，火爆的视频可能拥有100万条弹幕甚至更多。向已经“人满为患”的弹幕池里继续发送弹幕，是没有任何社交意义的：它们仅仅是居家隔离的人们感到憋得慌时，在互联网上发的几句用来找寻存在感的只言片语。很多时候，我们也极有可能被很多有偏见想法的弹幕所中伤，进而不得不屏蔽甚至关掉弹幕而错过了一些相对而言比较有营养、科普性的弹幕。

以上几种情景都不能算是线上社交，因为这些纷杂的弹幕对我们其实并不重要。人们总是以“重要性”来衡量什么是社交行为，而真正有意义的社交活动，往往是依托场景而存在。我和好友向弹幕池里发送的两条弹幕，在彼此间的意义微乎其微。而我们这个项目的出发点，正是如何让线上社交活动产生更多、更有价值的意义。

承接上文情境，如果说我们可以两个人，几个人的弹幕（**Bullet**），从茫茫弹幕人海中里抽离出来，并且保存收集好，放在一个共享的房间（**Room**）里面呢？如果是不光一个视频，要是所有的视频，甚至其他类型的资源（**Resource**），都可以按时间整理，并且放在对应组件（**Widget**）里，那么这些平时我们习以为常的线上互动行为，也会变得更具有社交的意义。归根结底这种意义的产生是依托于情境的变换，因为房间里的成员（**User**）是以不同资源为背景，创建独有的空间和记录独有的故事，从弹幕池中的分子变成了彼此的分母，而这也可以更好地表达了彼此之间的在意。

在此我们章鱼骑虎团队提出Bullet弹幕时间胶囊项目，由三个子项目构成，详情参见项目介绍。

## 项目介绍

我们的项目模型建立于提案中的故事背景。我们选择以弹幕时间胶囊Bullet平台作为切入点。Bullet主要由三个子项目构成：前端网页平台（[Bullet-Fronted](#)）、Chrome浏览器插件（[Bullet-Extension](#)）和后端API（[Bullet-Backend](#)）构成。

在建模的过程中，我们充分保留了延展的空间，并加入了好友（**Friend**），邀请（**Invitation**）和标签（**Tag**）模型，为以后的更多社交功能，平台趋势(trending tags)等数据分析、以及不同类型的房间类型提供了后端支持。

以下是我们在不同子项目使用到的技术栈

1. 前端：ReactJS, Apollo Client, SASS

2. 插件 : Bootstrap 4, Material UI, Preact, Vanilla JS, JQuery, SASS
3. 后端 : NodeJS, Apollo Server, Mongoose, GraphQL

## 应用场景

用户首次注册并登录Bullet网页端后，会进入到空空如也的主页。这时候用户可以选择创建房间（Room）并邀请其他用户（User）加入。房间创建完成后，可以进入房间界面添加资源（Resource）。

进行完网页平台操作后，便可以登录Chrome浏览器插件、选择该资源（由房间ID和资源ID共同定位）作为存储对象。

用户可以在浏览器插件中激活弹幕控制组件。当视频播放的同时，同房间内其他成员的弹幕会从页面中不同的高度飘过从右至左飘过。同时用户可以自己输入并发送弹幕（Bullet），届时会被存储在对应的资源中出现在其他人的视频中。

用户返回网页端房间（Room）并进入该资源（Resource）时，所有的成员弹幕会以时间线的形式展示出来。存储的所有弹幕可以根据房间、资源类型、发送者、时间进行过滤。

数据分析方面，在创建资源/发送弹幕的时候，可以选择性的给资源/弹幕加入标签（Tag）用于热度统计。

## 模型细节

以上应用场景的模型细节，因篇幅考虑放入以下补充技术文档中：

<https://github.com/qidian99/Bullet-Backend/blob/master/demo/README.md>。

## 小样演示

Youtube : <https://www.youtube.com/watch?v=hZTHzevIciY>

百度网盘: <https://pan.baidu.com/s/18YVDLqduLYAYH-DeOOtytg> 提取码: csjf

BiliBili : <https://www.bilibili.com/video/BV19A411E7GY/>

## 项目源码

Bullet网页端 ([Bullet-Fronted](#))

Bullet浏览器插件 ([Bullet-Extension](#))

Bullet后端 ([Bullet-Backend](#))