

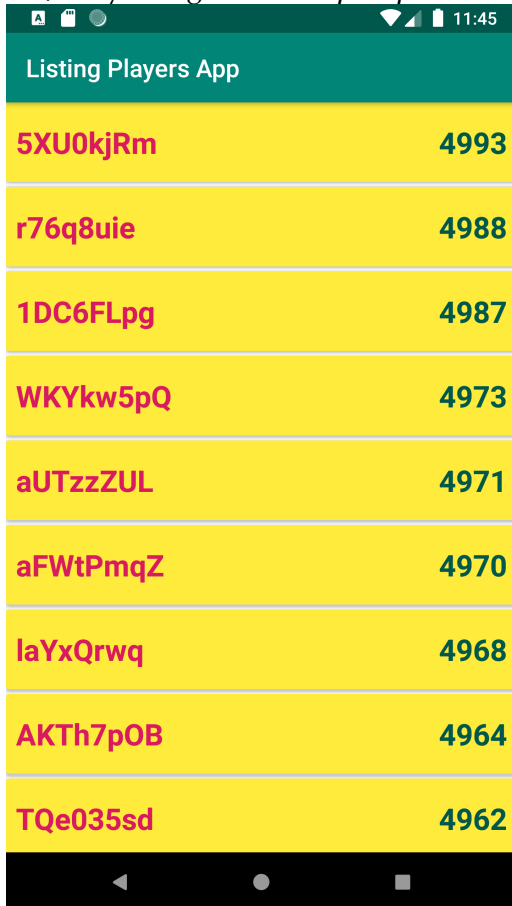
LẬP TRÌNH DI ĐỘNG (Tuần 09)

1. Viết API

- Lấy danh sách người chơi sắp xếp theo điểm chơi cao nhất giảm dần:
[/api/nguoi-choi](#)

2. Viết ứng dụng Listing Player App

- **MainActivity**: hiển thị danh sách người chơi (tên đăng nhập – điểm cao nhất) được lấy trong CSDL (sắp xếp theo điểm chơi cao nhất giảm dần)



The screenshot shows a mobile application titled "Listing Players App". It displays a list of players with their usernames and scores, sorted in descending order of score. The list contains 9 entries.

Username	Score
5XU0kJRm	4993
r76q8uie	4988
1DC6FLpg	4987
WKYkw5pQ	4973
aUTzzZUL	4971
aFWtPmqZ	4970
laYxQrwq	4968
AKTh7pOB	4964
TQe035sd	4962

- Kiểm tra kết nối mạng (Internet, Mobile data trước khi tải dữ liệu:

```
ConnectivityManager connMgr = (ConnectivityManager) getSystemService(Context.CONNECTIVITY_SERVICE);
NetworkInfo networkInfo = null;

if (connMgr != null) {
    networkInfo = connMgr.getActiveNetworkInfo();
}

if (networkInfo != null && networkInfo.isConnected()) {
    if (getSupportLoaderManager().getLoader(0) != null) {
        getSupportLoaderManager().restartLoader(0, data, this);
    }

    getSupportLoaderManager().initLoader(0, data, this);
} else {
    taoThongBao("Không thể kết nối internet").show();
}
```

- Tạo hộp thoại thông báo lỗi:

```
private AlertDialog taoThongBao(String s) {  
    // Tạo dialog  
    AlertDialog.Builder builder = new AlertDialog.Builder(this);  
    builder.setMessage(s).setTitle("Lỗi");  
    // Thêm nút "Đồng Ý" + xử lý sự kiện click  
    builder.setPositiveButton("Đồng Ý", new DialogInterface.OnClickListener() {  
        public void onClick(DialogInterface dialog, int id) {  
            // Thoát ứng dụng  
            moveTaskToBack(true);  
            android.os.Process.killProcess(android.os.Process.myPid());  
            System.exit(1);  
        }  
    });  
    // Hiển thị dialog  
    return builder.create();  
}
```

- Nếu dữ liệu trả về khi gọi API là null → thông báo **“Có lỗi khi truy vấn dữ liệu từ server...”**

```
if(data == null){  
    taoThongBao("Có lỗi khi truy vấn dữ liệu từ server...").show();  
    return;  
}
```

3. Cải tiến ứng dụng: Phân trang danh sách người chơi

- Ban đầu: hiển thị **25** người chơi
- Khi kéo xuống cuối danh sách → load thêm **25** người chơi tiếp theo từ CSDL vào danh sách (*khi chờ load danh sách thì hiển thị ProgressBar*)

Hướng dẫn:

- API: Thêm tham số vào QueryString: **api/nguoi-choi?page=1&limit=25**
(**page**: trang hiện tại, **limit**: số phần tử trên 1 trang)
Lưu ý: cần trả về tổng số trang hoặc tổng số dòng dữ liệu trong bảng
- Ứng dụng di động:
 - Thêm XML layout hiển thị ProgressBar
 - Cài đặt AsyncTaskLoader: thêm thuộc tính “trang hiện tại” và “số phần tử mỗi trang”
 - Cài đặt lại RecyclerView **Adapter**: phân biệt được ViewHolder: *hiển thị thông tin người chơi, hiển thị progressbar*
 - Cài đặt xử lý lắng nghe sự kiện **OnScrolled** của **RecyclerView**
 - Lấy tổng số phần tử trong danh sách, số phần tử được hiển thị trên RecyclerView, chỉ số của phần tử đầu tiên được hiển thị trên RecyclerView
 - Điều kiện để load thêm dữ liệu:
 - Không trong tình trạng load dữ liệu (isLoading = false)
 - Không phải là trang cuối cùng (isLastPage = false)
 - Số phần tử được hiển thị trên RecyclerView + Chỉ số của phần tử đầu tiên được hiển thị trên RecyclerView >= Tổng số phần tử trong danh sách
 - Chỉ số của phần tử đầu tiên được hiển thị trên RecyclerView >= 0
 - Tổng số phần tử trong danh sách >= Số phần tử trên mỗi trang (PAGE_SIZE)
 - Tải dữ liệu:
 - Trước khi gọi API để tải dữ liệu: thêm 1 phần tử **null** vào cuối danh sách phần tử trong RecyclerView → để hiển thị ProgressBar
 - Gọi API tải dữ liệu: truyền tham số “trang hiện tại” và “số phần tử mỗi trang”
 - Sau khi gọi API:
 - Bỏ phần tử **null** ở cuối danh sách phần tử trong RecyclerView → không hiển thị ProgressBar
 - Thêm các phần tử được trả về vào cuối danh sách phần tử trong RecyclerView
 - Kiểm tra đây có phải là “trang cuối cùng” không?