Proton Pack - User Test

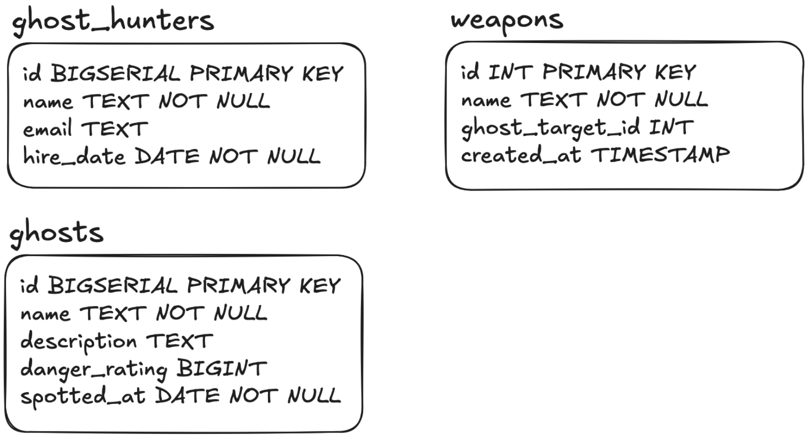
Introduction

Thank you for taking part in our user test. The goal of the user test is to evaluate the effectiveness of *Proton Pack* in regards of mitigating database migrations incidents. Therefore, users will be divided into two groups: one group being allowed to use *Proton Pack* and another group not allowed to use *Proton Pack*. Any other tool (e.g. web search, artificial intelligence) is available to both groups.

For this user test, you will jump into the role of a software engineer of Ghostbusters, Inc. and be responsible for database schema migration. You will receive an initial database schema and requirements describing potential schema changes. It is your task to deduct from the requirements how to change the database schema. There are multiple solutions to satisfy the requirements.

The schema changes should be proposed by submitting a .sql file. The file should include schema changes and NOT queries. **Users are encouraged to write down their considerations as comments**, as this will help to assess how users understand the requirements. Users should assume for this test, that they act in Ghostbusters, Inc.'s best interest. This means, that it's okay to implement a schema change and then add a comment on why performing this schema change might be a bad idea, besides it being a requirement.

Database Schema



ghost_hunters

```
id BIGSERIAL PRIMARY KEY
name TEXT NOT NULL
email TEXT
hire_date DATE NOT NULL
```

weapons

```
id INT PRIMARY KEY
name TEXT NOT NULL
ghost_target_id INT
created_at TIMESTAMP
```

ghosts

```
id BIGSERIAL PRIMARY KEY
name TEXT NOT NULL
description TEXT
danger_rating BIGINT
spotted_at DATE NOT NULL
```

Requirements

Product management has come up with some new ideas, that should be implemented. It is your task to fulfill the requirements by performing database schema changes. If you think requirements are stupid (e.g. implementing a requirement would result in data loss), you are free to leave comments for the different schema changes. If there is anything important, that you cannot implement right now due to various reasons, you can also mention it. **Both correct schema changes and critical considerations will positively influence your score.** The requirements are:

1. Users have reported that the overview screen is slow. The screen uses the ghosts table as data source. The table includes several million entries, some of them dating back to 1984. Since most of them irrelevant to most users, we propose to limit the screen to only show ghosts from 2020 or later.

2. From now on, it will be mandatory to provide an email address, when signing up as a ghost hunter in our app. This was optional before.

3. Due to new copyright claims by several weapon manufacturers, we must ensure that weapon names are unique in our system. This is an opportunity for our system, as it allows users to better search weapons by name. We want to have a new screen where users are allowed to search weapons by their name. This search should be lightning-fast.

4. We want to have a new screen, where data about a weapon is shown together with the target ghost (the ghost that this weapon is effective against). This should be fast, even though we must query two tables at once.

5. The overall database cost have exploded over the last months. We must save storage wherever we can! Can we optimize our existing tables that hold data?

Code

The initial database schema will be provided as a .sql file. The schema changes can be written into the same file.