A Third-party Bug tracking system provides an API endpoint **getIssues**, the request structure is like this:

```
{
  "project_id" : "project1"
}
```

- ***project_id*** *is the name of the project and is unique across the bug tracking system.*

The response structure is like this:

```
{
  "project_id" : "project1",
  "issues" : [
    {
      "issue_id" : "issue1",
      "type" : "bug",
      "current_state" : "open",
      "changelogs" : [
        {
          "changed_on" : "2017-01-01 12:00pm UTC",
          "from_state" : "open",
          "to_state" : "in_progress"
        },
        {
          "changed_on" : "2017-01-03 12:00pm UTC",
          "from_state" : "in_progress",
          "to_state" : "testing"
        },
        {
          "changed_on" : "2017-01-21 12:00pm UTC",
          "from_state" : "testing",
          "to_state" : "deploy"
        },
      ]
    },
    ...
  ]
}
```

- ***project_id*** *is the unique identifier for the project in system.*
- ***issues*** *is a list of issues filed for this project ever regardless of the state.*
- *Each issue can be uniquely identified across the project by* ***issue_id.***
- *Each issue has an associated* ***type*** *– which can be* ***bug***, ***enhancement*** *or* ***task***
- *Each issue has an associated* ***current_state*** *– which can be* ***open***, ***in_progress***, ***testing*** *or "****deploy****"*
- *Each issues has also a* ***changelogs*** *list – which is the set of state changes happened over time with a timestamp in sorted order. Oldest state change first, newest state change in last.*
- *The response time of bug tracking systems* ***getIssues*** *API is very slow – usually a couple of minutes.*

You have to design an API adapter which provides following API - **getWeeklySummary**:

- The API should be very fast – few milliseconds
- Should have near real-time data
- And should respect the fact that 3rd party bug tracking system might have some rate-limiting in place

The request structure should be like this:

```
{
  "project_id" : "project1",
  "from_week" : "2017W01",
  "to_week" : "2017W03",
  "types" : ["bug"],
  "states" : ["open"]
}
```

- ***project_id*** *is project name*
- ***from_week, to_week*** *represent week range. 2017W01 represents week one of 2017. Most of the modern programming languages provide API for converting YearWeek to dates. See for example: https://www.timeanddate.com/date/weeknumber.html*
- ***types*** *is the list of types of issues to be included in response, each item can be* ***bug****,* ***enhancement*** *or* ***task***
- ***states*** *is the list of states to be included in response, each item can be* ***open, in_progress****,* ***testing*** *or* ***deploy***

And the response should be like this:

```
{
  "project_id" : "project1",
  "weekly_summaries" : [
    {
      "week" : "2017W01",
      "state_summaries" :[
        {
          "state" : "open",
          "count" : 4,
          "issues" : [
            {
              "issue_id" : "issue1",
              "type" : "bug"
            },
            {
              "issue_id" : "issue2",
              "type" : "bug"
            },
            {
              "issue_id" : "issue3",
              "type" : "bug"
            },
            {
              "issue_id" : "issue4",
```

```
            "type" : "bug"
          }
        ]
      }
    ]
  },
  {
    "week" : "2017W02",
    "state_summaries" :[
      {
        "state" : "open",
        "count" : 2,
        "issues" : [
          {
            "issue_id" : "issue1",
            "type" : "bug"
          },
          {
            "issue_id" : "issue2",
            "type" : "bug"
          }
        ]
      }
    ]
  },
  {
    "week" : "2017W03",
    "state_summaries" :[
      {
        "state" : "open",
        "count" : 1,
        "issues" : [
          {
            "issue_id" : "issue1",
            "type" : "bug"
          }
        ]
      }
    ]
  }
 ]
}
```


- ***project_id*** *is project name*
- ***weekly_summaries*** *is a list of objects. Each object represents summary of a single week.*
- *Each* ***weekly_summary*** *object has a* ***week*** *field anda* ***state_summaries*** *list*
- ***state_summaries*** *is a list of objects. Each object represents a list of issues for that particular state.*
- *Each* ***state_summary*** *object has a* ***state*** *field,* ***count*** *of issues and a list of* ***issues****.*