Based on the assignment description you provided, here is a summarized checklist of the requirements you must meet to maximize your marks.

1. Software & Code Requirements (30 Marks)

- **Core Functionality:** Wrap a RESTful API into R functions.
- **Data Wrangling:**
  - Functions must output data in a **viable format** (e.g., data frames, lists), not raw binary (unless necessary).
  - *Constraint:* Only include general-use wrangling in the package functions. Specific/niche wrangling should be left for the vignette.
- **Error Handling:** Functions must handle API errors gracefully (don't just crash; provide useful error messages).
- **Cross-Platform:** The package must compile and install on **Windows, Mac, and Linux**.
- **Testing:** You must implement both **unit tests** and **integration tests**.
- **Best Practices:** Code must follow standard R API package best practices (likely referring to `httr` or `httr2` style guides).
- **Internal Documentation:** Code must have inline comments for developers, not just user-facing help files.

2. Documentation & Vignettes (50 Marks)

- **Vignette (Crucial):**
  - Create at least one vignette illustrating most or all of the package's functionality.
  - **Hard Requirement:** The vignette must demonstrate a **"Zero to Graph"** workflow. You **must** include a visualization (plot) derived from the API data.
- **User Documentation:** All functions must be fully documented (likely using `roxygen2`).
- **Readability:** Documentation must be clear, complete, and accessible to a new user.

3. Workflow & Process (Essential for Evaluation)

- **Collaboration Method:**
  - Use a shared GitHub repository.
  - **NO Direct Commits:** You must work via **Pull Requests** (using forks or branches).
  - Commit and push often.
- **Daily Notebook:** Keep a daily project notebook (logs of activity/decisions).
- **Continuous Integration (CI):**
  - Use **Travis CI** (or a similar CI tool if permitted, but the prompt specifically requests Travis).
  - The build must pass.
  - Include a **passing build stamp (badge)** in your `README.md`.

- **Git History:** Must demonstrate that the workflow defined in your `CONTRIBUTING.md` was actually followed and that all members contributed equally.

4. Mandatory Repository Files Checklist

Ensure your repository contains these specific files:

- [ ] `README.md` (with CI badge).
- [ ] `LICENSE` (describing usage permissions).
- [ ] `CODE_OF_CONDUCT.md` (behavioral expectations).
- [ ] `CONTRIBUTING.md` (outlining workflow/how to contribute).
- [ ] `vignettes/` (containing your analysis and plot).
- [ ] `tests/` (containing your unit/integration tests).

5. The "Bonus" Path

- **Go Public/Open Source:** If the project is a "good idea" (even if not perfectly polished) and you make it open to the public, you are eligible for bonus points.