

Product Requirements Document (PRD)

Product Name: Heroku CLI Integration with Visual Studio

Overview:

The Heroku CLI Integration with Visual Studio is a feature that allows developers to manage their Heroku applications directly from within Visual Studio. This integration aims to simplify the development and deployment process for Heroku users, making it easier to create, deploy, and manage applications on the Heroku platform.

Product Goals:

1. **Simplify the development and deployment process:** Provide a seamless integration between Visual Studio and Heroku, allowing developers to manage their applications without leaving the IDE.
2. **Improve productivity:** Reduce the time and effort required to deploy and manage Heroku applications, enabling developers to focus on writing code.
3. **Enhance user experience:** Provide a user-friendly interface for managing Heroku applications, making it easier for developers to get started with Heroku.

Functional Requirements:

1. Heroku CLI Installation

- **Description:** The Heroku CLI should be installable from within Visual Studio.
- **Acceptance Criteria:**
 - The Heroku CLI is installed successfully.
 - The installation process is straightforward and easy to follow.
- **Assumptions and Dependencies:**
 - The user has a valid Heroku account.
 - The user has the necessary permissions to install the Heroku CLI.

2. Heroku Login and Authentication

- **Description:** Users should be able to log in to their Heroku account from within Visual Studio.

- **Acceptance Criteria:**
 - The user is able to log in successfully.
 - The login process is secure and follows best practices for authentication.
- **Assumptions and Dependencies:**
 - The user has a valid Heroku account.
 - The user has the necessary permissions to log in to their Heroku account.

3. App Management

- **Description:** Users should be able to create, deploy, and manage Heroku applications from within Visual Studio.
- **Acceptance Criteria:**
 - The user is able to create a new Heroku application.
 - The user is able to deploy their application to Heroku.
 - The user is able to manage their application's configuration and settings.
- **Assumptions and Dependencies:**
 - The user has a valid Heroku account.
 - The user has the necessary permissions to create and manage Heroku applications.

4. Git Integration

- **Description:** The Heroku CLI integration should support Git version control.
- **Acceptance Criteria:**
 - The user is able to push changes to their Heroku application using Git.
 - The integration supports Git branching and merging.
- **Assumptions and Dependencies:**
 - The user has a valid Git repository.
 - The user has the necessary permissions to push changes to their Heroku application.

5. Error Handling and Debugging

- **Description:** The integration should provide clear and concise error messages and debugging information.

- **Acceptance Criteria:**
 - The integration provides clear and concise error messages.
 - The integration provides debugging information to help the user diagnose and resolve issues.
- **Assumptions and Dependencies:**
 - The user has a valid Heroku account.
 - The user has the necessary permissions to view error messages and debugging information.

Non-Functional Requirements:

1. Performance

- **Description:** The integration should not significantly impact the performance of Visual Studio.
- **Acceptance Criteria:**
 - The integration does not slow down Visual Studio.
 - The integration does not consume excessive resources.
- **Assumptions and Dependencies:**
 - The user has a valid Heroku account.
 - The user has the necessary permissions to use the integration.

2. Security

- **Description:** The integration should follow best practices for security and authentication.
- **Acceptance Criteria:**
 - The integration uses secure authentication protocols.
 - The integration stores sensitive information securely.
- **Assumptions and Dependencies:**
 - The user has a valid Heroku account.
 - The user has the necessary permissions to use the integration.

3. Usability

- **Description:** The integration should provide a user-friendly interface that is easy to navigate and understand.
- **Acceptance Criteria:**
 - The integration provides clear and concise documentation.
 - The integration provides an intuitive user interface.
- **Assumptions and Dependencies:**
 - The user has a valid Heroku account.
 - The user has the necessary permissions to use the integration.

Success Metrics:

4. **Adoption Rate:** The number of users who install and use the Heroku CLI integration.
5. **User Satisfaction:** The level of satisfaction among users who use the Heroku CLI integration.
6. **Error Rate:** The number of errors encountered by users who use the Heroku CLI integration.

Open Issues:

7. **Heroku CLI version compatibility:** The integration may not be compatible