

Contents

1	Project Drivers	2
1.1	The Purpose of the Project	2
1.2	The Stakeholders	2
1.3	Mandated Restraints	2
1.4	Naming Conventions and Terminology	3
1.5	Relevant Facts and Assumptions	3
2	Functional Requirements	3
2.1	The Scope of the Work and the Product	3
2.2	The Context of the Work	3
2.3	Work Partitioning	3
2.4	Individual Product Use Cases	3
2.5	Functional Requirements	3
3	Non-functional Requirements	3
3.1	Look and Feel Requirements	3
3.2	Usability and Humanity Requirements	3
3.3	Performance Requirements	3
3.4	Operational and Environmental Requirements	3
3.5	Maintainability and Support Requirements	4
3.6	Security Requirements	4
3.7	Cultural Requirements	4
3.8	Legal Requirements	4
4	Project Issues	4
4.1	Open Issues	4
4.2	Off-the-Shelf Solutions	4
4.3	New Problems	4
4.4	Tasks	4
4.5	Migration to the New Product	4
4.6	Risks	4
4.7	Costs	4
4.8	User Documentation and Training	4
4.9	Waiting Room	4
4.10	Ideas for Solutions	4

1 Project Drivers

1.1 The Purpose of the Project

PyCards is a collection of solitaire (single-player) card games. It is designed to be a source of entertainment for the end user(s). The main objective for the product is to PyCards to be a user-friendly application that our users can use to pass the time, have fun, relax, and also challenge themselves.

1.2 The Stakeholders

The Client
The Customers
Other Stakeholders

1.3 Mandated Restraints

Constraint 1

The source code of the application shall be implemented using the Python programming language, version 2.7.xx.

Rationale 1

The product should preserve compatibility for existing users of the original implementation. The client should not be required to upgrade or install new software in order to run the application.

Fit Criterion

The product shall only require a Python interpreter (and/or the inclusion of additional Python packages) in order to operate. Exception will be made if either the entirety of the product is ported to another language or if the necessary runtimes and dependencies can be bundled with the product.

Constraint 2

The product shall be compatible with Windows operating systems (Windows 10), Mac OSX, and Ubuntu provided that Python version 2.7.xx is compatible with the target system.

Rationale 2

The client should not be required to migrate to newer (or different) software or hardware in order to utilize the product.

Fit Criterion

The product shall be tested on each of the specified systems by the developers to ensure that it operates as expected and is fully functional.

Constraint 3

The product shall be made available under the GNU GPLv3 or later, along with any non-permissive terms added in accord with section 7 of the GNU GPLv3.

Rationale 3

The original implementation was conveyed with this license and as per the conditions modified source versions must be licensed under the same license.

Fit Criterion

The product must adhere to the conditions of the GNU GPLv3 license.

Constraint 4

The source code for the product shall be publicly available but the end product must also be deliverable as an standalone executable or application for both Windows operating systems and Mac OSX operating system.

Rationale 4

The source code for the product must be publicly available as per the conditions of the license. Users

should not be required to be familiar with the command-line/terminal in order to operate the product.

Fit Criterion

The final product should be available as a .exe executable for Windows-based systems and as an application for OSX systems. The code should be available in a publicly accessible repository.

1.4 Naming Conventions and Terminology

Source code:

Compatibility:

Implementation:

Product:

1.5 Relevant Facts and Assumptions

Relevant Facts

- The existing implementation is implemented purely in the Python programming language.
- Mac OSX and Ubuntu come with a Python environment pre-installed. The installed version may or may not be compatible with the product.

Assumptions

- It is not feasible for the product to be tested for compatibility on all of the various operating systems. As such it will only be tested on the latest version of each major operating system as previously specified.
- The product assumes that the system that it will be operated on includes peripherals such as a mouse, keyboard and display.

2 Functional Requirements

2.1 The Scope of the Work and the Product

2.2 The Context of the Work

2.3 Work Partitioning

2.4 Individual Product Use Cases

2.5 Functional Requirements

3 Non-functional Requirements

3.1 Look and Feel Requirements

- The product shall represent cards with images of playing cards and additional areas that can be interacted with (ex. where cards can be placed) shall be outlined for the user to see.

3.2 Usability and Humanity Requirements

- The product shall be easy for a child with basic reading and computer abilities to use

3.3 Performance Requirements

- Normal interaction actions with the game shall take no longer than if an average user were playing the same game with a physical deck of cards.

3.4 Operational and Environmental Requirements

- Users will interact with the product using a mouse and keyboard connected to their computer

3.5 Maintainability and Support Requirements

- New MIS reports
- The product is expected to run on Windows 10, Mac OSX, and Ubuntu systems that have Python version 2.7.xx installed.

3.6 Security Requirements

3.7 Cultural Requirements

- The product will not contain images or text that would be considered offensive to residents of North America.

3.8 Legal Requirements

- The product is licensed under the GNU GPLv3 license and must conform to the license and any non-permissive terms added in accordance to section 7 of the GNU GPLv3 license.

4 Project Issues

4.1 Open Issues

4.2 Off-the-Shelf Solutions

4.3 New Problems

4.4 Tasks

4.5 Migration to the New Product

4.6 Risks

4.7 Costs

4.8 User Documentation and Training

4.9 Waiting Room

4.10 Ideas for Solutions

Appendix

Symbolic Parameters

List of Figures