(SRS)

Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <03/05/10> | <1.0> | Đặc tả lại toàn bộ yêu cầu | Lê Văn Long |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Giới thiệu 4

1.1 Mục đích 4

1.2 Phạm vi 4

1.3 Các định nghĩa và các từ viết tắt 4

1.4 Tham khảo 4

1.5 Tổng quan 4

2. Mô tả tổng quan 4

2.1 Triển vọng sản phẩm 4

2.2 Những chức năng chính của sản phẩm 4

2.3 Đặc thù người dùng 5

2.4 Những actor chính 5

2.5 Những ràng buộc chung 5

2.6 Những giả định và phụ thuộc 5

3. Những yêu cầu 5

3.1 Về mặt chức năng 5

3.1.1 Chức năng học chính tả 5

3.1.2 Chức năng tập đọc 5

3.1.3 Chức năng luyện từ và câu 6

3.1.4 Chức năng kể chuyện 6

3.1.5 Chức năng tập làm văn 6

3.1.6 Chức năng xem đáp án 6

3.1.7 Chức năng chơi game 6

3.1.8 Chức năng viết nhật ký 6

3.1.9 Chức năng giải đố vui 6

3.1.10 Chức năng đọc chuyện cổ tích 6

3.2 Usability 6

3.2.1 <Usability Requirement One> 7

3.3 Reliability 7

3.3.1 <Reliability Requirement One> 7

3.4 Performance 7

3.4.1 <Performance Requirement One> 8

3.5 Supportability 8

3.5.1 <Supportability Requirement One> 8

3.6 Design Constraints 8

3.6.1 <Design Constraint One> 8

3.7 On-line User Documentation and Help System Requirements 8

3.8 Purchased Components 8

3.9 Interfaces 8

3.9.1 User Interfaces 8

3.9.2 Hardware Interfaces 9

3.9.3 Software Interfaces 9

3.9.4 Communications Interfaces 9

3.10 Licensing Requirements 9

3.11 Legal, Copyright, and Other Notices 9

3.12 Applicable Standards 9

4. Supporting Information 9

# Giới thiệu

## Mục đích

SRS mô tả đầy đủ các hành vi của ứng dụng. Đồng thời nó cũng mô tả các yêu cầu phi chức năng, những ràng buộc thiết kế và những nhân tố cần thiết khác để tạo nên một bản mô tả đầy đủ và toàn diện yêu cầu phần mềm.

## Phạm vi

SRS này áp dụng cho phần mềm Tiếng Việt 4 do các nhóm 16, 17, 18, 19, 20 phát triển. Đi kèm với tài liệu này là tài liệu về mô hình [Use-Case](4.Use%20Case%20Suite.doc).

## Các định nghĩa và các từ viết tắt

## Tham khảo

Phần mềm [Em học tốt lớp 4](Yeucautuphanmemtuongtu/Yêu%20cầu%20tham%20khảo%20từ%20phần%20mềm%20“Em%20học%20tốt%20lớp%204”%20(nhóm%20phát%20triển%20phần%20mềm%20sinh%20viên%20học%20sinh%20SSDG).pdf) do nhóm phát triển phần mềm sinh viên học sinh phát hành.

## Tổng quan

Phần còn lại của SRS được tổ chức như sau: Phần 2 mô tả một cách tổng quan về phần mềm. Nó chỉ ra những chức năng chính mà phần mềm có thể làm được, những ràng buộc chung và một số giả định.Phần 3 đưa ra những yêu cầu cụ thể của phần mềm. Phần 4 đưa ra những mở rộng trong tương lai có thể có của phần mềm.

# Mô tả tổng quan

## Triển vọng sản phẩm

Về cơ bản, phần mềm sẽ giúp học sinh lớp 4 học các bài học và làm bài tập trong sách giáo khoa Tiếng Việt 4. Với giao diện vui nhộn, ngộ nghĩnh và dễ sử dụng, phần mềm rất phù hợp với lứa tuổi của học sinh lớp 4.

Phầm mềm được phát triển dưới dạng stand-alone, không phụ thuộc vào các sản phẩm phần mềm khác. Có thể chạy được trên nền windows.

## Những chức năng chính của sản phẩm

|  |  |  |
| --- | --- | --- |
| Loại Use-case | Use-case | Mô tả Use-case |
| Use-case liên quan đến việc học tập | Học tập đọc | Học các bài tập đọc |
| Luyện chính tả | Luyện tập về chính tả, thường dưới dạng điền vào chỗ trống |
| Kể chuyện | Người dùng nghe nội dung câu chuyện và trả lời câu hỏi |
| Luyện từ và câu | Người dùng đọc nội dung đoạn văn bản, rút ra các nhận xét, chương trình đưa ra các ghi nhớ, sau đó người dùng áp dụng kiến thức vừa học được để làm bài tập |
| Tập làm văn | Tham khảo các dàn ý và các bài văn mẫu, soạn thảo bài tập làm văn |
| Xem đáp án | Xem đáp án các bài tập |
| Use-case liên quan đến việc giải trí | Chơi game | Các trò chơi vui nhộn |
| Viết nhật ký | Viết nhật ký cá nhân |
| Giải đố vui | Trả lời các câu đố vui |
| Đọc chuyện cổ tích | Đọc các câu chuyện cổ tích có trong phần mềm |

## Đặc thù người dùng

* Người dùng chủ yếu là học sinh lớp 4, chưa có kiến thức về tin học, thích các hình ảnh ngộ nghĩnh, vui nhộn.

## Những actor chính

Actor chính là “người dùng”.

## Những ràng buộc chung

TiengViet4 là phần mềm đơn người dùng (chỉ có một người dùng tại một thời điểm).

## Những giả định và phụ thuộc

Các trò chơi trong TiengViet4 là các trò chơi mã nguồn mở trên mạng Internet, không phải do nhóm phát triển viết ra.

# Những yêu cầu

## Về mặt chức năng

Danh sách các chức năng dưới đây được sắp theo độ ưu tiên.

### Chức năng học chính tả

Mã chức năng: F-001

Use-case liên quan đến chức năng học chính tả: UC-04

### Chức năng tập đọc

Mã chức năng: F-002

Mô tả: Phần mềm sẽ hiển thị nội dung bài đọc với những hình ảnh minh họa, cùng với phần giải thích ngữ nghĩa của những từ khó. Phần cuối bài đọc sẽ có danh sách những câu hỏi dưới dạng trắc nghiệm để kiểm tra lại kiến thức học sinh thu được qua bài đọc.

Use-case liên quan đến chức năng học chính tả: UC-03

### Chức năng luyện từ và câu

Mã chức năng: F-003

Mô tả: Phần mềm hiển thị nội dung bài đọc và danh sách các câu hỏi gợi ý để người dùng tự rút ra các nhận xét. Sau đó phần mềm sẽ hiển thị nội dung các ghi nhớ và phần nội dung luyện tập tương ứng. Nội dung luyện tập có thể lấy từ sách giáo khoa, hoặc là câu hỏi trắc nghiệm do nhóm phát triển biên soạn, căn cứ theo sách giáo khoa.

Use-case liên quan đến chức năng học chính tả: UC-05

### Chức năng kể chuyện

Mã chức năng: F-004

Mô tả: Chương trình phát ra đoạn âm thanh kể về một câu chuyện. Sau đó người dùng sẽ trả lời các câu hỏi trắc nghiệm liên quan nội dung câu chuyện và xem kết quả.

Use-case liên quan đến chức năng học chính tả: UC-06

### Chức năng tập làm văn

Mã chức năng: F-005

Mô tả: Chương trình hiển thị một đề bài tập làm văn và một cửa sổ cho người dùng soạn thảo văn bản. Người dùng có thể xem dàn ý và bài làm mẫu của đề bài đó.

Use-case liên quan đến chức năng học chính tả: UC-07

### Chức năng xem đáp án

Mã chức năng: F-006

### Chức năng chơi game

Mã chức năng: F-007

### Chức năng viết nhật ký

Mã chức năng: F-008

### Chức năng giải đố vui

Mã chức năng: F-009

### Chức năng đọc chuyện cổ tích

Mã chức năng: F-010

## Usability

[This section includes all those requirements that affect usability. For example,

* specify the required training time for a normal users and a power user to become productive at particular operations
* specify measurable task times for typical tasks or base the new system’s usability requirements on other systems that the users know and like
* specify requirement to conform to common usability standards, such as IBM’s CUA standards Microsoft’s GUI standards]

### <Usability Requirement One>

[The requirement description goes here.]

## Reliability

[Requirements for reliability of the system should be specified here. Some suggestions follow:

* Availability—specify the percentage of time available ( xx.xx%), hours of use, maintenance access, degraded mode operations, and so on.
* Mean Time Between Failures (MTBF) — this is usually specified in hours, but it could also be specified in terms of days, months or years.
* Mean Time To Repair (MTTR)—how long is the system allowed to be out of operation after it has failed?
* Accuracy—specifies precision (resolution) and accuracy (by some known standard) that is required in the system’s output.
* Maximum Bugs or Defect Rate—usually expressed in terms of bugs per thousand lines of code (bugs/KLOC) or bugs per function-point( bugs/function-point).
* Bugs or Defect Rate—categorized in terms of minor, significant, and critical bugs: the requirement(s) must define what is meant by a “critical” bug; for example, complete loss of data or a complete inability to use certain parts of the system’s functionality.]

### <Reliability Requirement One>

[The requirement description.]

## Performance

[The system’s performance characteristics are outlined in this section. Include specific response times. Where applicable, reference related Use Cases by name.

* Response time for a transaction (average, maximum)
* Throughput, for example, transactions per second
* Capacity, for example, the number of customers or transactions the system can accommodate
* Degradation modes (what is the acceptable mode of operation when the system has been degraded in some manner)
* Resource utilization, such as memory, disk, communications, and so forth.

### <Performance Requirement One>

[The requirement description goes here.]

## Supportability

[This section indicates any requirements that will enhance the supportability or maintainability of the system being built, including coding standards, naming conventions, class libraries, maintenance access, and maintenance utilities.]

### <Supportability Requirement One>

[The requirement description goes here.]

## Design Constraints

[This section indicates any design constraints on the system being built. Design constraints represent design decisions that have been mandated and must be adhered to. Examples include software languages, software process requirements, prescribed use of developmental tools, architectural and design constraints, purchased components, class libraries, and so on.]

### <Design Constraint One>

[The requirement description goes here.]

## On-line User Documentation and Help System Requirements

[Describes the requirements, if any, for o-line user documentation, help systems, help about notices, and so forth.]

## Purchased Components

[This section describes any purchased components to be used with the system, any applicable licensing or usage restrictions, and any associated compatibility and interoperability or interface standards.]

## Interfaces

[This section defines the interfaces that must be supported by the application. It should contain adequate specificity, protocols, ports and logical addresses, and the like, so that the software can be developed and verified against the interface requirements.]

### User Interfaces

[Describe the user interfaces that are to be implemented by the software.]

### Hardware Interfaces

[This section defines any hardware interfaces that are to be supported by the software, including logical structure, physical addresses, expected behavior, and so on.]

### Software Interfaces

[This section describes software interfaces to other components of the software system. These may be purchased components, components reused from another application or components being developed for subsystems outside of the scope of this **SRS** but with which this software application must interact.]

### Communications Interfaces

[Describe any communications interfaces to other systems or devices such as local area networks, remote serial devices, and so forth.]

## Licensing Requirements

[Defines any licensing enforcement requirements or other usage restriction requirements that are to be exhibited by the software.]

## Legal, Copyright, and Other Notices

[This section describes any necessary legal disclaimers, warranties, copyright notices, patent notices, wordmark, trademark, or logo compliance issues for the software.]

## Applicable Standards

[This section describes by reference any applicable standard and the specific sections of any such standards which apply to the system being described. For example, this could include legal, quality and regulatory standards, industry standards for usability, interoperability, internationalization, operating system compliance, and so forth.]

# Supporting Information

[The supporting information makes the **SRS** easier to use. It includes:

* Table of contents
* Index
* Appendices

These may include use-case storyboards or user-interface prototypes. When appendices are included, the **SRS** should explicitly state whether or not the appendices are to be considered part of the requirements.]