



English Wordtrain with Superman

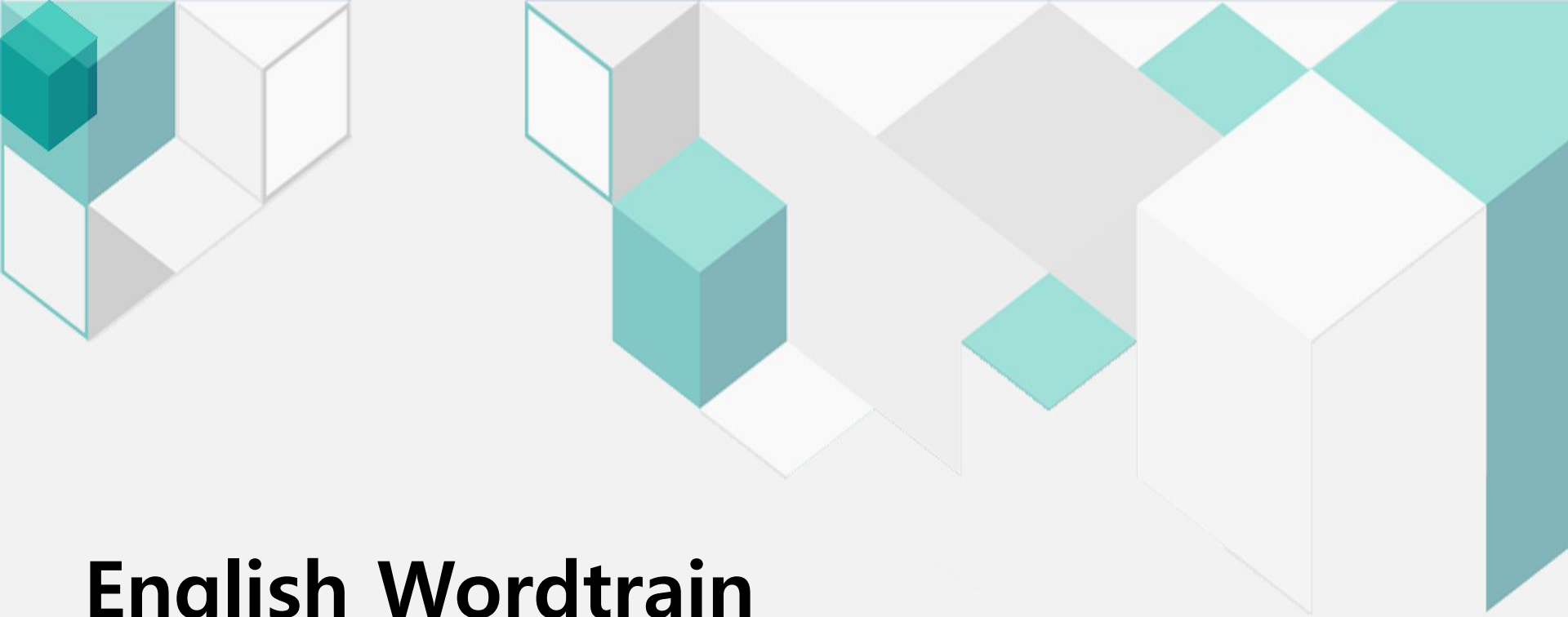
OSP 3rd Cycle Demonstration

<Team 4>

201111354 박정민
201111356 박준한
201111384 정국빈

Index

- ◆ **English Wordtrain with Superman**
- ◆ **Co-work with Software Verification**
- ◆ **About OSP**
- ◆ **Epilogue**



English Wordtrain with Superman

3rd Cycle 수정사항

- 경험치 반영

Category	Brute Force Testing Report
Problem	Wordtrain Game에서 얻은 경험치가 Parent Mode에 반영이 안됨
Solution	GUI를 수정하는 과정에서 코드의 누락 발생 : 코드 수정으로 해결

- 그림 파일 수정 (like)



English Wordtrain with Superman

- Why Superman???

-> 예능 프로그램인 '슈퍼맨이 돌아왔다'에서 채용
(아버지를 '슈퍼맨'이라고 칭함)

- 요즘 부모님과 아이들간의 소통이 중요시되기 때문에 부모님과 소통하는 프로그램 개발이 목적이었다.

Main GUI



Account

PARENT PHD

1234

CHILD NAME

KB

CONFIRM

ENTER

BACK

ESC

Parent Mode

SEND MESSAGE 1

GOAL SETTING 2

CHECK GOAL 3

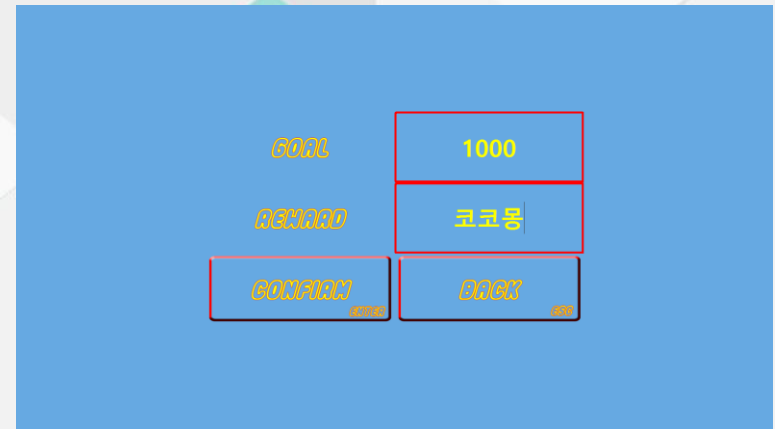
RESET 4

BACK 5

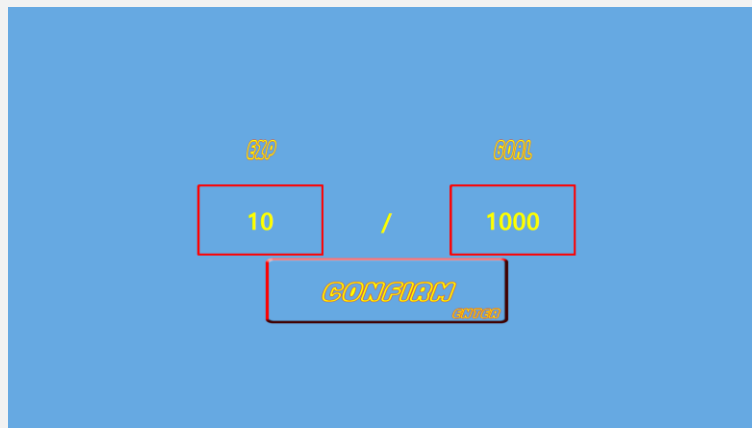
Parent Mode (cont.)



↑ Send Message



↑ Goal Setting



← Check Goal

Child Mode

NAME **KB** *LEVEL* **1** *EXP* **10**

GOAL **1000** *REWARD* **코코몽**

MESSAGE **밥 먹어라**

PRACTICE 1

GAME 2

DICTIONARY 3

BACK 4

Wordtrain Practice

WORDTRAIN PRACTICE

talk

1

thank

2

there

3



SHIFT

NEXT

SPACE

BACK

ESC

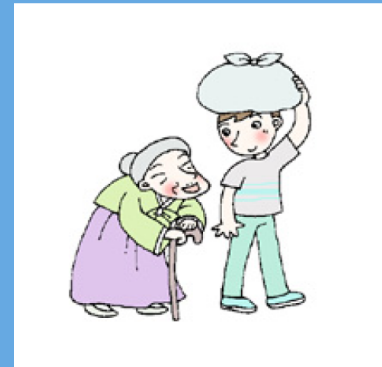
school



long



goat




감사하다

Wordtrain Game

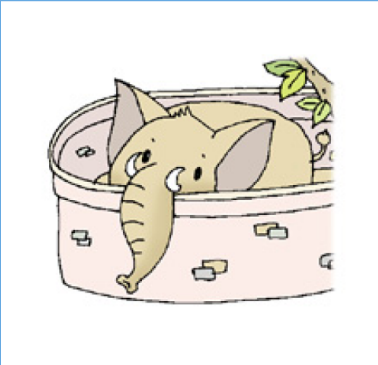
WORDTRAIN GAME

elephant

 *SHIFT*

NEXT *SPACE*

BACK *ESC*



코끼리

⇒ zoo ⇒ orange

Dictionary

DICTIONARY

food



SHIFT

NEXT

SPACE

BACK

ESC



음식



Co-work with Software Verification

Software Verification Team

- 소프트웨어 검증 4조와 진행
 - 박현규, 최정한, 김민우, 함진아
- 평소에 친분이 있던 팀이기 때문에 연락을 하기가 수월했다.
- 하지만 테스트를 할 때 매우 자세한 부분까지 검증을 하여서 많은 부분을 수정하게 하였다.

Advantage of Co-work

- 개발하면서 모르고 지나치는 부분들을 잡아주기 때문에 버그를 수정하는 데 많은 도움이 되었다.
- 여러 문서간의 통일성을 높여주어서 처음 생각했던 방향으로 개발을 진행하게 길을 잡아주었다.
- 테스트를 다른 팀이 해주므로 개발에만 좀 더 집중을 할 수 있었기 때문에 전체적인 진도를 빨리 나갈 수 있었다.

Disadvantage of Co-work

- 전체적인 문서 양식이 통일이 되지 않아서 처음 문서를 받았을 때 파악하는 데 어려움이 있었다.
- 전체적인 개발 환경을 시작 전에 확실히 협의를 못했기 때문에 개발할 때 어려움이 있었다.



About OSP

Advantage of OSP

- 처음에 작성한 제안서와 설계서에 맞게 프로그램을 개발하기 때문에 체계적인 개발이 가능하다.
- 각 단계마다 대응되는 부분이 있기 때문에 다음 단계를 설계하는데 용이하다.
- 일정 단계의 Cycle을 반복하므로 프로그램의 지속적인 수정이 가능하다.

Disadvantage of OSP

- 단계별로 진행해서 결과물을 도출하는데 하위 단계의 오류나 수정사항을 발견 시에 그 단계부터 위 단계까지 모든 문서를 검토해야 된다.
- 설계에 너무 많은 시간을 들이다 보면 개발하는 데 시간이 모자라는 경우가 생긴다.
- 개발 팀과 검증 팀의 소통이 제대로 되지 않아서 서로의 요구사항을 이해 못하다 보면 제대로 된 개발이 불가능해지기 쉽다.

Improvement Plan of OSP

- 각종 다이어그램의 검증을 개발 단계가 아니라 문서 단계부터 진행한다면 나중에 모든 문서를 한 번에 수정하는 경우가 줄어들 것이다.
- 검증 팀과 협의 후에 검증 문서 양식도 통일된다면 테스트 결과를 이해하기 더 수월할 것이다.



Epilogue

Epilogue

- 설계부터 시작해 바로 구현하지 않고, 먼저 전체적인 부분의 틀을 만들 수 있는 게 매력적이었다. 하지만 당연히 그 틀은 틀린 부분이 많아 고칠게 많은 게 조금 부담되었다. 그리고 확실히 우리 조가 아닌 다른 사람의 눈으로 문서나 프로그램을 검증해 우리가 찾지 못했던 부분이나, 쉽게 지나쳐간 부분들을 바로 잡아주고, 그 팀과 소통하는 방법을 배운 것 같아서 좋았다.
- 전체 프로그래밍을 계획 설계 구현 까지 해본 것은 처음이라 계획 단계에서 처음에는 조금 헤매었지만 설계 및 구현 단계에서는 우리가 계획한대로 진행하는 것이라 색다른 경험이었다. 그리고 검증 팀과의 소통은 좋은 경험이었다. 우리 시선이 아닌 다른 팀의 시선으로 우리 프로그램을 평가 받아서 우리가 찾지 못한 수정사항도 발견하여서 좋았다. 다음에 나도 검증 수업을 들어서 검증 팀의 입장도 경험해 보고 싶다



Thank You !

Q&A

