



OSP 3rd Cycle

Subject : Software Modeling

Professor : 유준범

T1 | 201111341 김성민

201111379 이한빈

201111397 황정아

OUTLINE

I. Revision

II. Program Development

III. OSP

IV. OSP Review

V. Demonstration

I. Revision

I. Revision

- Playing sound from the internet, instead of playing sound from files. (To minimize the size of exe file)

II. Program Development

Why conversation based English Learning program?

- > Easily become familiar with English
- > More effective learning than memorization studying

II. Program Development

Functional Requirements:

- Child Mode: studying words, studying simple conversation (random) display sound, display sentences or words with pictures.
- Parent Mode: Find word, Add word, automatically display sound

Non-Functional Requirements:

- Children should feel interested through our program
- Conversations should be educational
- UI should be suitable for children
- Reading/Writing from Database should be fast

II. Program Development

How we fulfilled Requirements

- Minimize the size of exe file by playing sound from internet
- Adjusting difficulties by adding words from Parents Mode
- Pictures suitable for kids (animation images instead of real photo)
- More conversation contents can allow higher level of listening and speaking

II. Program Development


Main



II. Program Development

Parent Mode

[대화형]영어교육 프로그램_★★★Talkids★★★



WORD	MEAN
aquarium	수족관
camel	낙타
chicken	닭
crap	개
crocodile	악어
dog	개
dolphin	돌고래
door	문
duck	오리
eagle	독수리
elephant	코끼리
fox	여우
frog	개구리
giraffe	기린
goose	가위
leopard	표범
lion	사자
monkey	원숭이
ostrich	타조
otter	수달
penguin	펭귄
pigeon	비둘기
rabbit	토끼
raccoon	너구리
shark	상어
snake	뱀
sparrow	참새
swan	백조
whale	고래

monkey

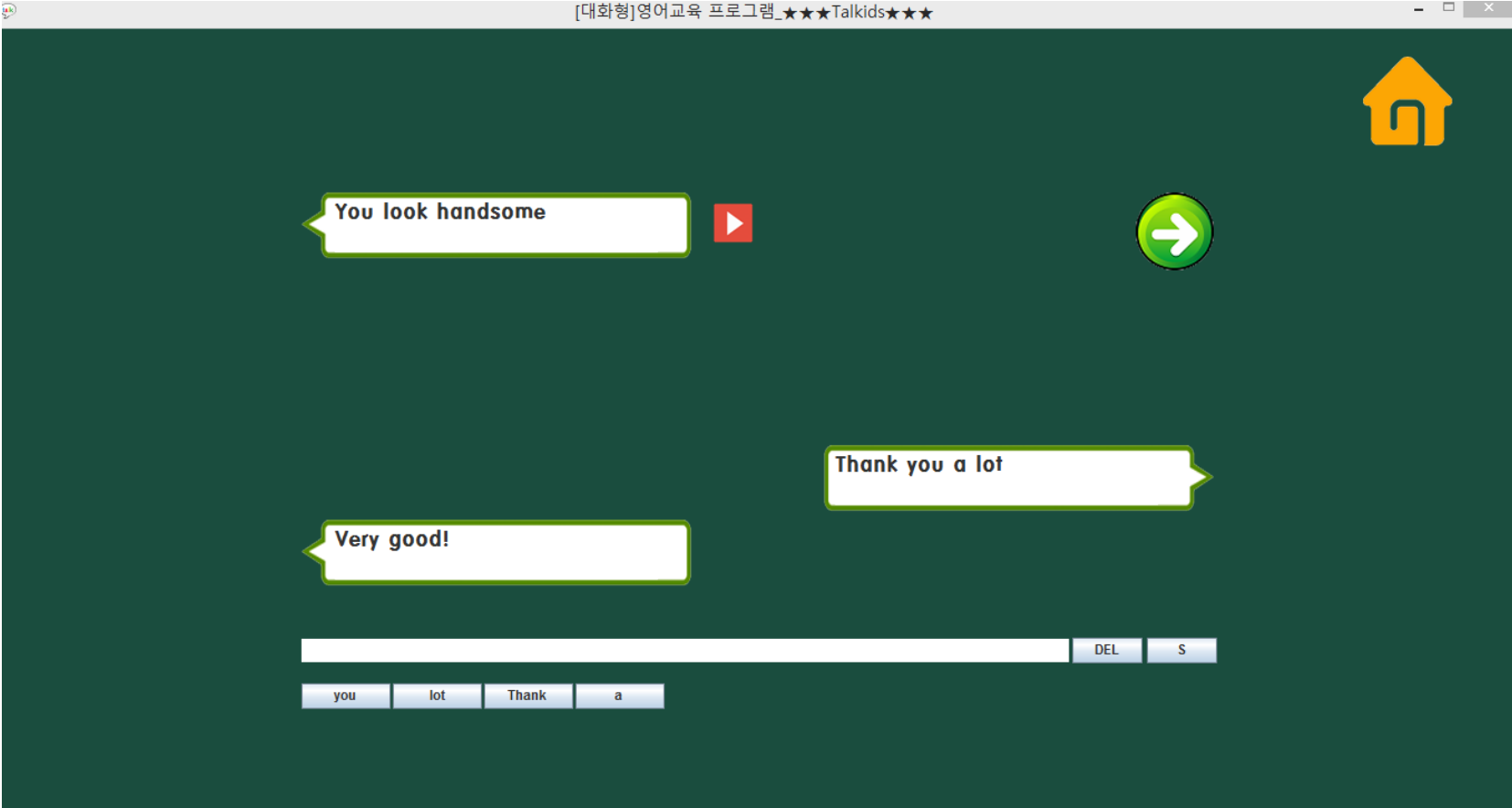
II. Program Development

Child Mode



II. Program Development

Child Mode



III. OSP

III. OSP

Positive aspects of OSP

- Appropriate for big projects with many developers involved
- Easy to understand the project through documentations.
- Easy to program if documentation is well-constructed
- After model structure is set(ex. Unit testing) it can be used repeatedly
- Program come close to perfection through each cycle

III. OSP

Negative aspects of OSP

- To add more functions through cycles, all documentations have to be checked and edited.
- Not suitable for small size of program.

Improvements for OSP

- > adjusting the size of cycle depending on the size of project.
- > Tool to compare and analyze documentations

IV. Review

IV. Review

Working with Software Verification team

Positive aspects:

- Advices from the seniors
- more efficient programming was available through unnecessary source code founded with detailed time and statistics,

Negative aspects:

- Lack of communication with SV team delayed our project
- Test case made by SV team was too small, therefore some of the testing were done by ourselves.
- After updating redmine and git, communication lacked (Some of the features were tested without noticing the update)
- Some of the testing reports from SV team was not specified (ex. "Moving from main to child mode or parents mode take very long time")

IV. Review

Epilogue

- Using software development process is efficient in many ways.
- Verification and Validation from SV team was quite hard but very helpful
- There is no right answer in software development
- Using external tools to develop diagrams was a great experience.
- Output from each stage is an input for next stage, it was easy but hard
- Synchronizing documentation and source code was one of the difficulty

V. Demonstration

Q&A ?

THANK YOU