App Inventor For Android: Graffiti App

Lyn Turbak, Wellesley College
Reading Virtual Walls

Phone 1

Graffiti

Pick Wall  start  Visit Wall

Post a message here.

Post! 

Walls:

Post a message here.  This is a test wall

start  test
Writing Virtual Walls

Phone 1

Graffiti

Pick Wall | start | Visit Wall

lyn says hi

Post a message here.

Phone 2

Graffiti

Pick Wall | start | Visit Wall

This is a test wall

Walls:

lyn says hi

Post a message here.

This is a test wall
Creating Virtual Walls

**Phone 1**

- **Graffiti**
  - **Pick Wall**
  - **start**
  - **Visit Wall**
  - **Post!**

  lyn says hi
  ------------------------------
  Post a message here.

**Phone 2**

- **Graffiti**
  - **Pick Wall**
  - **AIA**
  - **Visit Wall**
  - **Post!**

  lyn says hi
  ------------------------------
  Post a message here.

**Walls:**

- **start**
  - lyn says hi
  ------------------------------
  Post a message here.

- **test**
  - This is a test wall

- **AIA**
  - I like App Inventor
The Wall named *walls* is Special
Can Select a Wall via Pick Wall Menu

Phone 1

Graffiti
AIA
start

Phone 2

Graffiti
Pick Wall  AIA  Visit Wall

I like App Inventor

Walls:

lyn says hi

-----------------
Post a message here.

start

This is a test wall
test

I like App Inventor

AIA
Try it Out!

- Projects > Graffiti
- Package for Phone > Show Barcode
- Read barcode via Barcode Scanner
- Play with the app.
Graffiti Web Service: Getting a Value

```
getValue
  tag: test
```

```
["VALUE", "test", "This is a test wall"]
```

---

**Wall Database**

```
I like App Inventor
```

---

**Graffiti Web Service API**

**Post a message here.**

- Lyn says hi
- This is a test wall
- I like App Inventor
**Graffiti Web Service: Storing a Value**

```
storeavalue
tag: test
Value: "kilroy was here"
```

```
["STORED", "test", "kilroy was here"]
```

**Graffiti Web Service API**

**Wall Database**

- lyn says hi
- kilroy was here
- This is a test wall
- I like App Inventor

---

Google App Engine
Interacting with the Graffiti Web Service

1. Via Browser:
   Visit  http://lyng-graffiti.appspot.com/

2. Via AIA TinyWebDB Component:
   Will see how to do this in a few slides.
Graffiti Components

Components

Screen1
- WallSelectionArrangement
  - WallPicker
  - WallNameTextBox
- PostArrangement
  - PostTextBox
  - PostButton
- WallContents
- WallDB
- Clock

Graffiti

Pick Wall
Visit Wall
Post!

Hidden components

TinyWebDB instance
Graffiti Blocks: Three Parts

1. Getting contents of wall specified in text box.
2. Posting to the wall specified in text box.
3. Implementing Pick Wall button.
Graffiti Blocks: Three Parts

1. Getting contents of wall specified in text box.
2. Posting to wall specified in text box.
3. Implementing Pick Wall button.
Getting a Wall: Calling the visitWall Procedure

- When VisitWallButton.Click
  - Call visitWall wallArg WallNameTextBox.Text

- When Clock:Initialize
  - Call visitWall wallArg text start
Getting a Wall: Defining the visitWall Procedure

def wallName

text

to visitWall arg name wallArg arg
do set global wallName to value wallArg
set WallNameTextBox.Text to value wallArg
if test not global wallName = text
then-do set WallContents.Text to global wallName join text : looking up contents ...
call updateWall
Getting a Wall: the updateWall Procedure
Getting a Wall: Timer-based Updates

![Diagram of timer-based updates](image)

**Properties: Clock**

- **ID**: Clock
- **TimerInterval**: 10000
- **TimerEnabled**: Enabled
- **TimerAlwaysFires**: Disabled
Graffiti Blocks: Three Parts

1. Getting contents of wall specified in text box.
2. Posting to wall specified in text box.
3. Implementing Pick Wall button.
Posting to a Wall

Key statement: store message in DB

Update wall contents after message stored.
Graffiti Blocks: Three Parts

1. Getting contents of wall specified in text box.
2. Posting to wall specified in text box.
3. Implementing Pick Wall button.
Picking a Wall: Looking up Wall Names

Initially the empty string; will become a comma-separated list of wall names, such as “AIA,berlin,start”

Additional code to look up wall names

Additional code to process wall names
Picking a Wall: ListPicker Methods
Other Ways to Use TinyWebDB

- Hal Abelson’s simple database: e.g.,
  - store lists of words for Mad Libs
  - give English translations for Spanish words
  - associate countries names with flags

- Ellen Spertus’s radio client

- Dave Wolber’s Amazon client

- Eni Mustafaraj’s voting app
Wellesley’s CS114 Socio-Techno Web Class

- Taught by Takis Metaxas (lecture) and Eni Mustafaraj (lab instructor). I was “evangelist”.

- 17 students; 12 did apps for final project

- Sample projects:
  - Rumor generator
  - Interactive Wellesley Tour
  - Got Places?
  - Who’s Keeping Score?
Extensions

Some things to try:

- Include email address of sender
  (use EmailUserAddress of a Voting component)

- Use BarcodeScanner component to “read”
  wall names (a la Sharon Perl’s Wallz app).

- Tutorials and sample applications at
  [http://sites.google.com/site/appinventorhelp](http://sites.google.com/site/appinventorhelp)