



Unified Butterfly Recorder

butterflies.ece.iastate.edu | butterflies@iastate.edu | Dec13-08



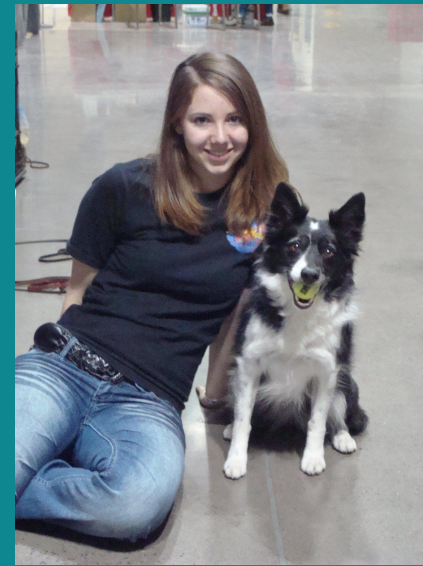
Team Butterfly



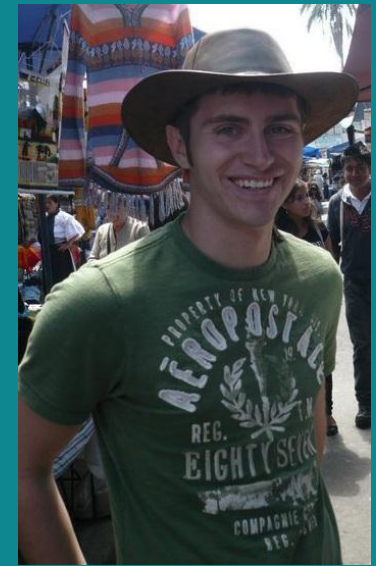
Cameron
Whipple



Curtis
Ullerich



Julie
Tillman



Ryan
Scheel

Seniors, Computer Engineering



Team Butterfly



Nathan
Brockman



Anita
Westphal



Dr. Diane
Rover

Reiman Gardens

IOWA STATE UNIVERSITY
Department of Electrical and Computer Engineering



Background

- Butterflies are a great indicator species
 - Climate change
 - Habitat restoration progress
- Different survey protocols
- Varied surveyor experience



Problems

- Protocols designed to answer specific questions
 - Disparate storage and collection formats
 - No common storage location or method
- Large-scale observations are hard



Goals

- Create an app for data collection during butterfly field surveys
- Streamline and standardize
- Centralize data

"This app will help standardize the collection of data and has the potential of impacting conservation efforts both nationally and globally."

- Anita Westphal, Reiman Gardens



Timeline (All dates are 2013)

- Mar 1 Survey responses aggregated & requirements set
- Mar 15 Determine location of database for this summer
- Apr 1 Divide tasks among team members; begin prototype
- May 10 Finish Android prototype with ability to store data in the server; beta release
- Aug 15 Aggregate feedback from field testing
- Sept 15 All feedback from summer addressed; have iOS prototype finished
- Oct 15 All known bugs fixed; version 1.0 release
- Dec 10 Poster and presentation finalized



Work Breakdown

- Team
 - Research, planning, requirements
- Management Roles
 - Documents and website: Curtis
 - Database: Cameron
 - Prototype: Ryan
 - Presentation and communication: Julie



Deliverables

- Android and iOS app usable on both tablets and phones
- Database
- Web interface for record access



Market and Literature Survey

Questionnaire

Butterfly Survey App Questionnaire

Responses to this survey will guide the specification for an Android/iOS app intended to make conducting butterfly surveys in the field easier and more standard. Thank you for your valuable input. Contact butterflies@iastate.edu with any questions.

* Required

Name *

Email *

Organization

How often do you perform a Pollard survey? *

Examples: never, every week, 3 times per year

How often do you perform a Mark Recapture survey? *

Examples: never, every week, 3 times per year

How often do you perform a Distance Sampling survey? *



Market and Literature Survey

- Market feedback: questionnaire
 - International response
 - Butterflies and Moths of North America (BAMONA)
 - Use results to generate requirements
- Existing apps
- Review standards for existing survey protocols

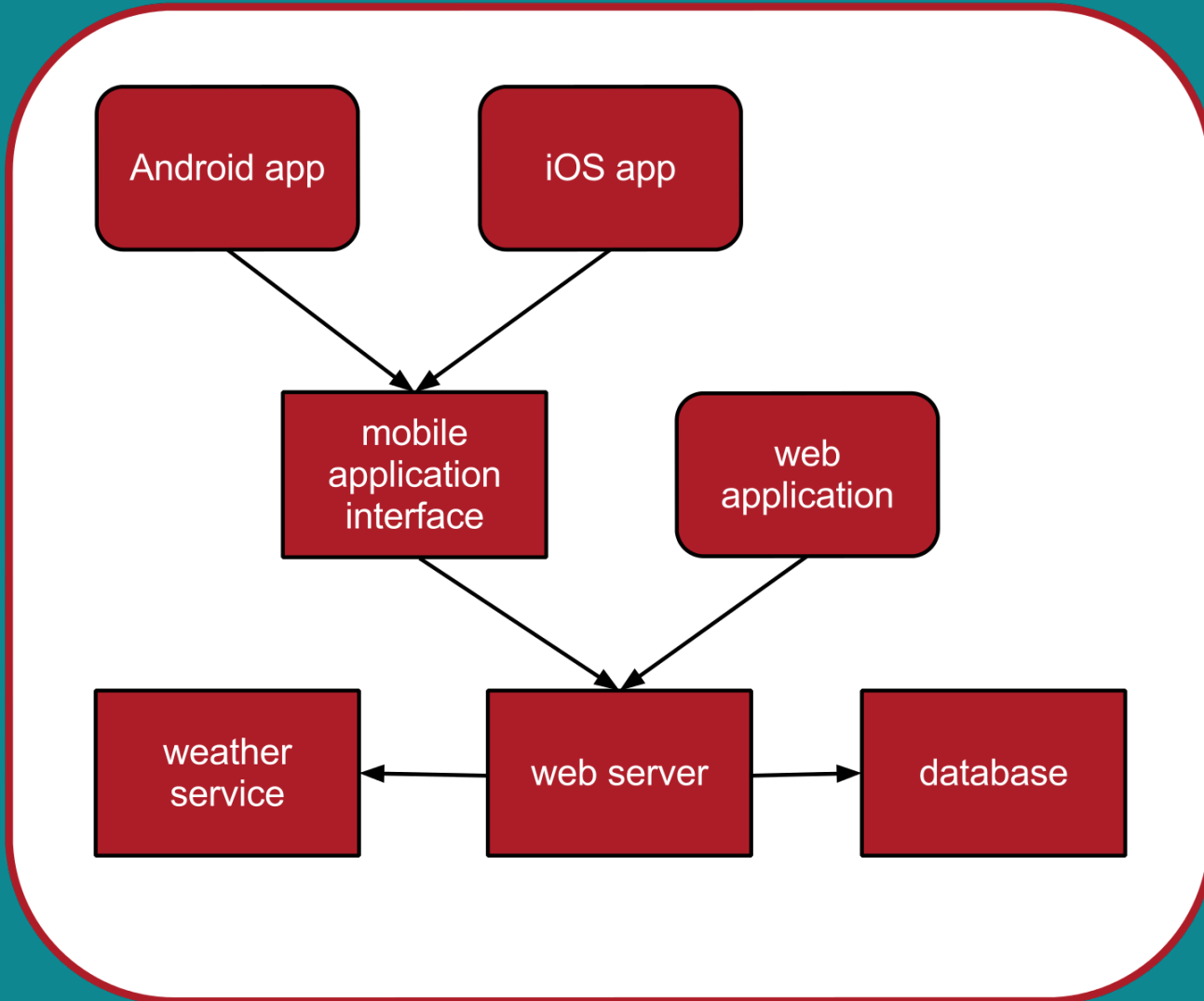


System Requirements

- Functional
 - Input data points on app
 - Import weather data
 - Export through web interface
- Nonfunctional
 - Performance
 - Usability
 - Security



Architecture





Database

- Merge our schema with BAMONA's
- Export through existing web interface
- Local storage on mobile device
- MySQL

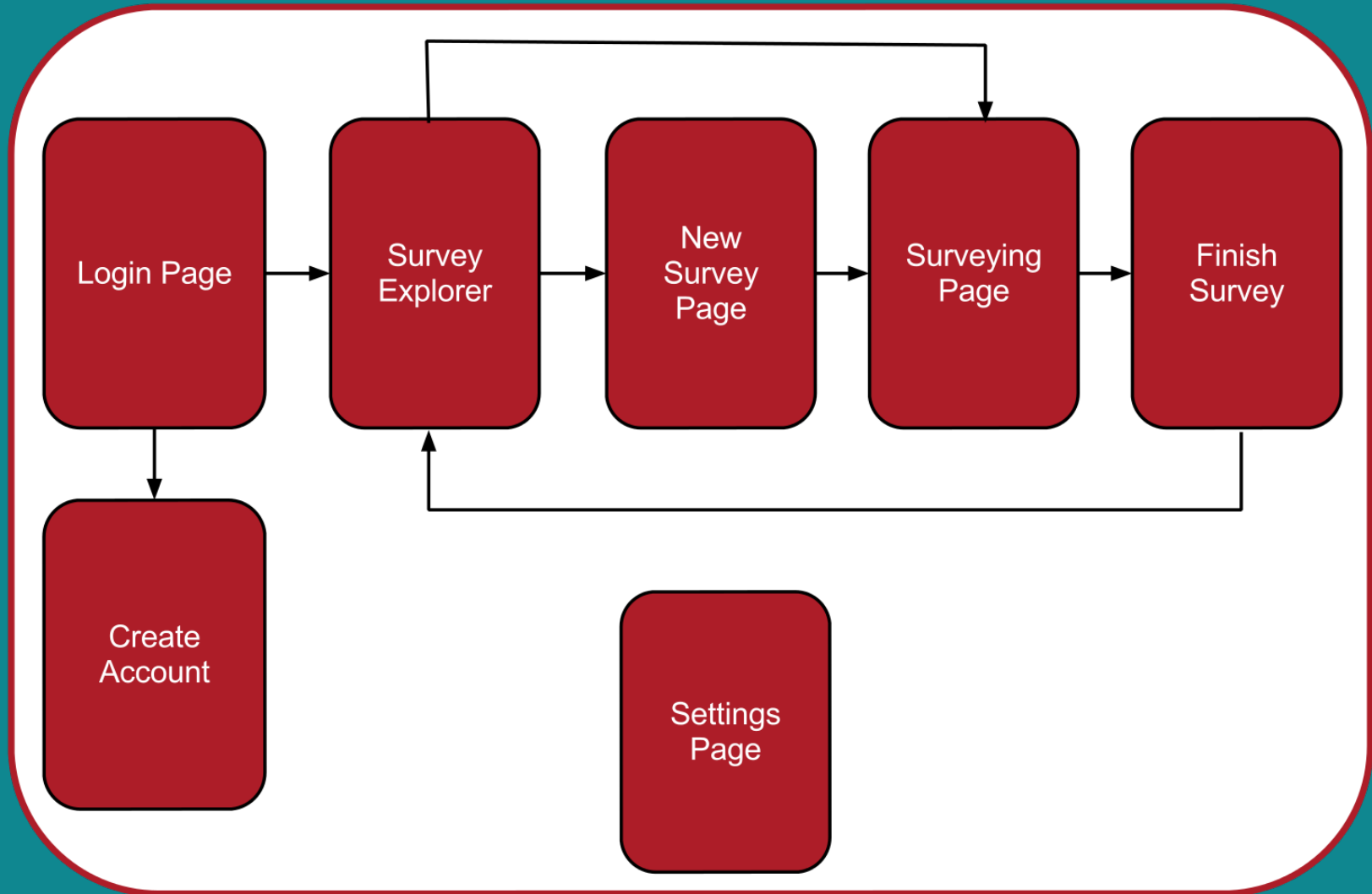


Web Server

- Co-design, BAMONA will host
- Pull in weather data
- Host web interface
- Provide interface to database
- Drupal
- XML-RPC



Screenflow Diagram





Unified Butterfly Recorder

Enter your credentials to continue.

Email

Password

Log in to UBR

Create an account

Create an Account

Enter your e-mail address and choose a password.

Email

Password

Confirm Password

Create account

Create Survey

Enter Survey Information

Survey Name

Site Name

Choose observer ID

100

Survey Type

Meandering

Create



User Interface Design

- User selects protocol at the time of the survey
- Collect data in the background for all protocols
- Goal: Collect a superset of the data no matter what protocol is used
- Optimize survey screen for speed



Prototype

- Complete Android UI with one protocol
- Feedback feature
- Submission to server and local export
- Release on Google Play Store



Testing

- Software unit/regression testing
 - Android testing framework/JUnit and Xcode Unit Test target
- Code coverage
 - EclEmma and Xcode
- Field testing
- Verification and validation



Risks

- iOS Application
- Data license
- Loss of 3rd party service
- Battery life
- Loosely defined original requirements



Progress So Far

- Talked with several butterfly professionals
- Secured collaborator for database
- Prototype in progress
- Client and advisor are happy!



Near Future

- Prototype release
- Finalize database schema with BAMONA
- Integrate weather service
- Address blocking bugs during summer
- Do surveys ourselves



Next Semester

- Collect feedback and adjust design
- BAMONA account integration
- Android and iOS version 1.0 release with all protocols and requirements
- Determine long-term maintenance plan



Impact

- Real-world release and benefits
- Help answer conservation questions
- Innovative interface
- BAMONA = faster market penetration, better maintainability



Acknowledgements

- Nathan and Anita
- Dr. Rover
- Thomas Naberhaus, BAMONA lead developer



Questions or comments?

butterflies.ece.iastate.edu | butterflies@iastate.edu



Interface Descriptions

- Phone → Database
 - All data sent to database using XML-RPC commands of the Drupal API.
- Web server → Database
 - The web application will communicate with the database using Drupal's database abstraction layer in PHP.
- Weather service → Web server
 - Weather service abstraction layer through which we periodically get weather information from a 3rd party.



Database Schema (1) (rough sketch)

Main Table

key - Record_ID

Record_ID -> Links to

Environment and Pictures

Surveyor_ID -> Links to Surveyor
Information

Survey_ID

Start Time

End Time

Site_Name / Route_Name

Longitude - GPS

Latitude - GPS

Amount_of_Butterfly_Seen

Weather Service

Environment Table

key - Record_ID

Record_ID

Temperature

Wind_Speed

Wind_Direction

Cloud_Cover

Pictures Storage

- key - Record_ID

Record_ID

path to photo



Database Schema (2) (rough sketch)

User Environment Table

key - Record_ID

Record_ID

Habitat_Type / Category

Section_Number

Temperature

Wind_Speed

Wind_Direction

Light_Level

Cloud_Cover

Comments

Surveyor Table

key - Record_ID

Surveyor_ID

Surveyor_Name

Surveyor_Email

Surveyor_PasswordHash

Surveyor_Phone

Surveyor_Organization

Surveyor_Comments



Functional Requirements (1)

- Create a default list of butterfly species based on location; this list must be customizable by the user
- Users can download data from the database
- Weather data must be imported from an existing service



Functional Requirements (2)

- Data Points:

- Collected/Calculated Automatically

- GPS
 - waypoint of routes

- date and time
 - start/stop time

- default species list
 - light levels

- distance/habitat category
 - walking pace

- differentiation of sections along route

- Input By User

- tally of sightings for each species
 - habitat category

- habitat conditions
 - data entry for mark recapture details

- behavior notes
 - site name

- level of ID certainty
 - surveyor information

- Web Services

- taxonomic tree-based classification
 - misc. comment section

- temperature
 - wind speed

- % cloud cover
 - wind direction



Nonfunctional Requirements

- Performance
- Ease of use
- Security
- Graceful failure
- Form factor adaptability
- Offline usability
- Hardware adaptability
- Minimal data transfer
- Network agnosticism
- Transactional data submissions
- Multi-task capability
- Minimal resource usage
- Security



Weather Services

- **Iowa Environmental Mesonet**
 - Service hosted by Department of Agronomy (ISU)
 - Access to historical data only
 - More complicated to use and integrate within application/server
 - <http://mesonet.agron.iastate.edu/request/download.phtml>
- **World Weather Online**
 - Free access to current weather information but a fee for historical data
 - Nice API that would be easy to use and integrate within application/server
 - <http://www.worldweatheronline.com/>

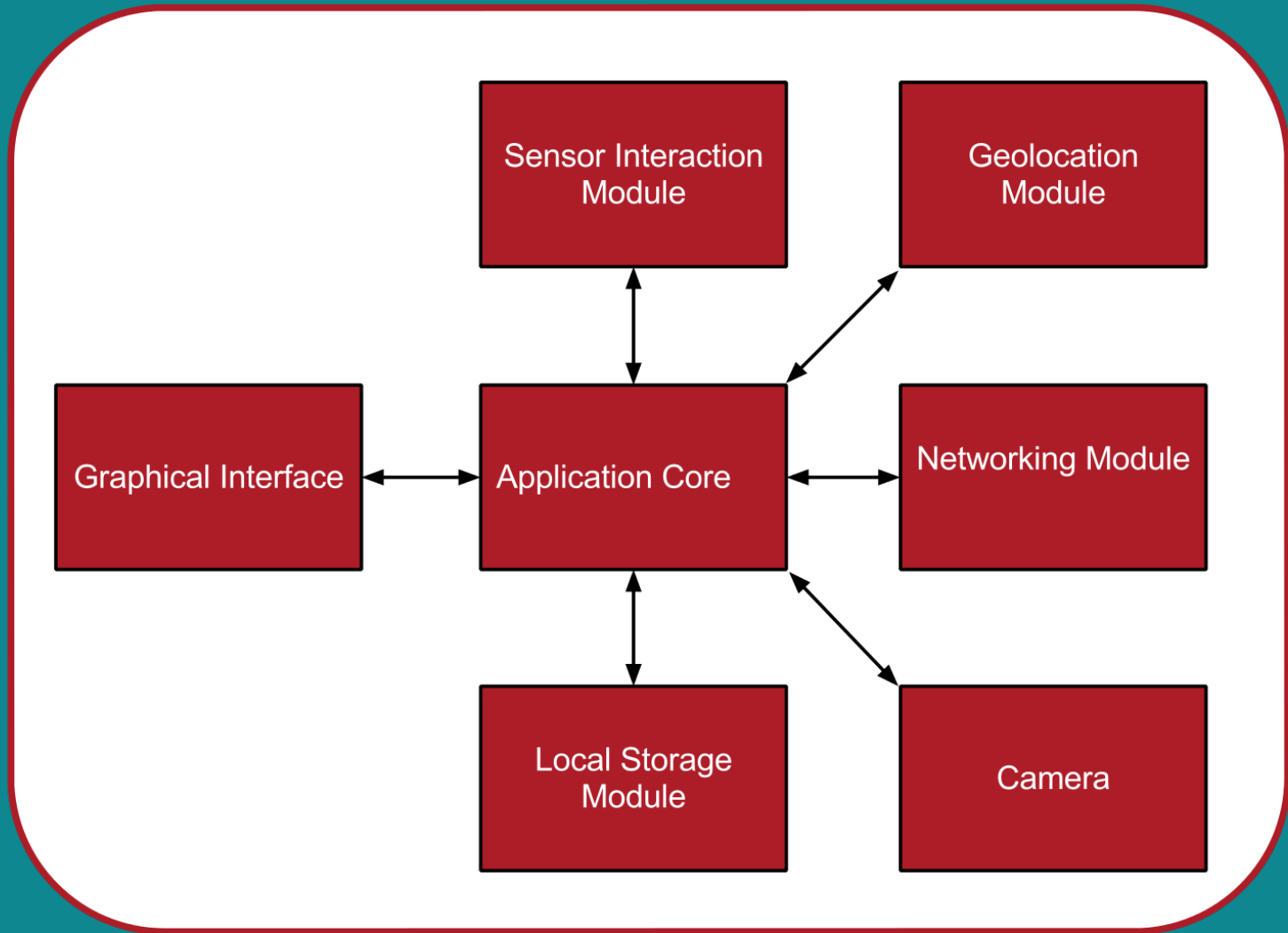


BAMONA

- Thomas Naberhaus, lead developer
- Co-designing database and web interface
- Host and maintain database
- Implement web interface
- High impact



App Architecture





Backup Server

- MySQL
- Apache 2.2.3 / PHP
- Redhat 5