



Chrome:

Face First

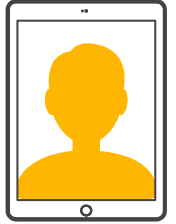
Bits... Please!

Agenda:

1. Face First
2. Approaches
3. Final Approach
4. Sequence Diagram
5. Impact on Architecture
6. Testing Implementation
7. Current Limitations/
Lessons Learned



Face First

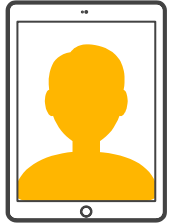


Problem:

- Users have many passwords associated with numerous websites
- Difficult to remember all passwords
- Security

Solution:

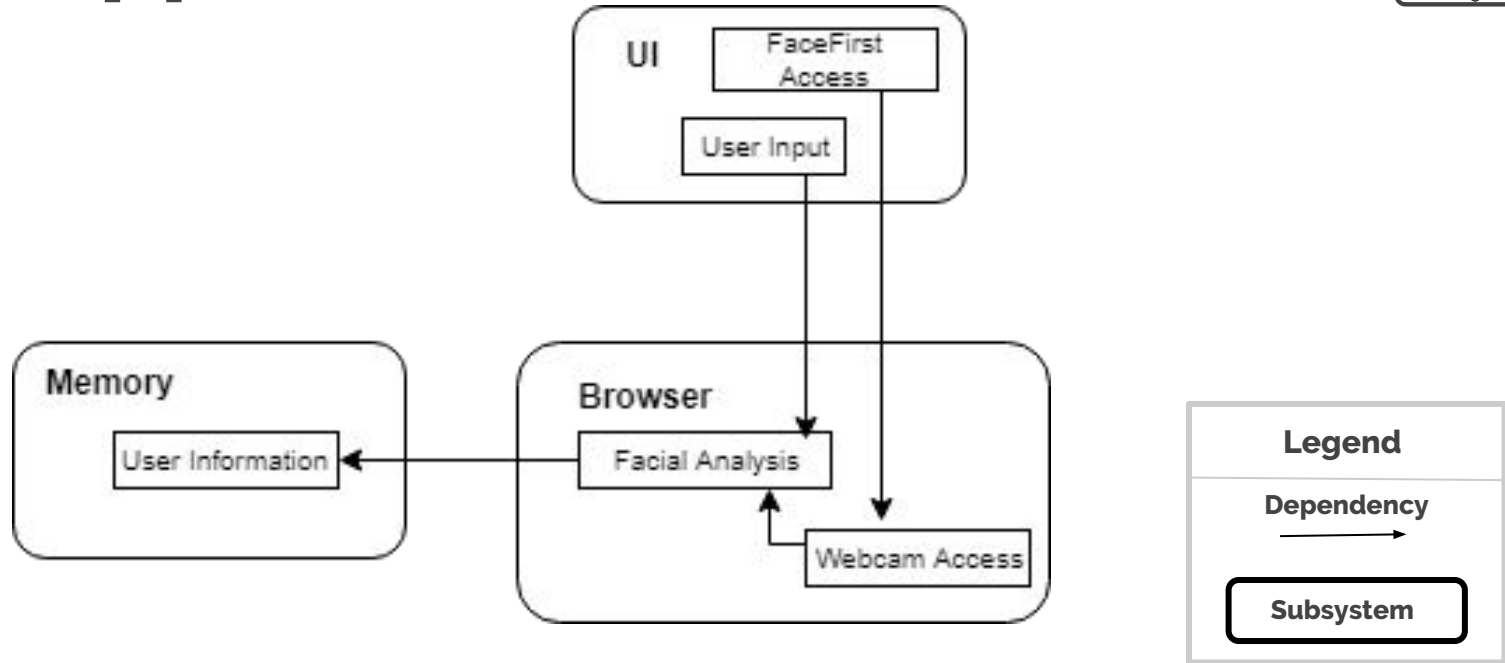
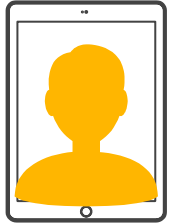
- Add your face authentication to Chrome
- Passwords are saved to the user account
- Use facial authentication to access user account



1st Approach

- Login valid for entire window
- Each tab opened in window automatically logged in
- Automatically logged out when window is closed

1st Approach





SAAM Analysis

Stakeholders

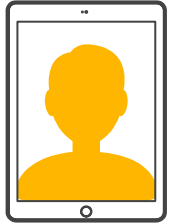
- Users
- Developers
- Investors
- Google

Pros

- Speed
- Lower coupling
 - Less subsystems involved in implementation

Cons

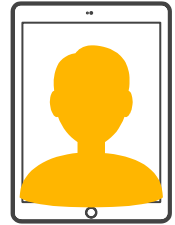
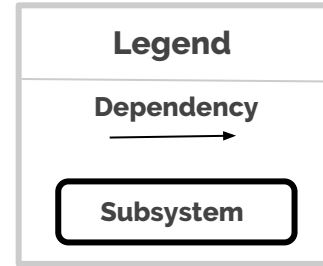
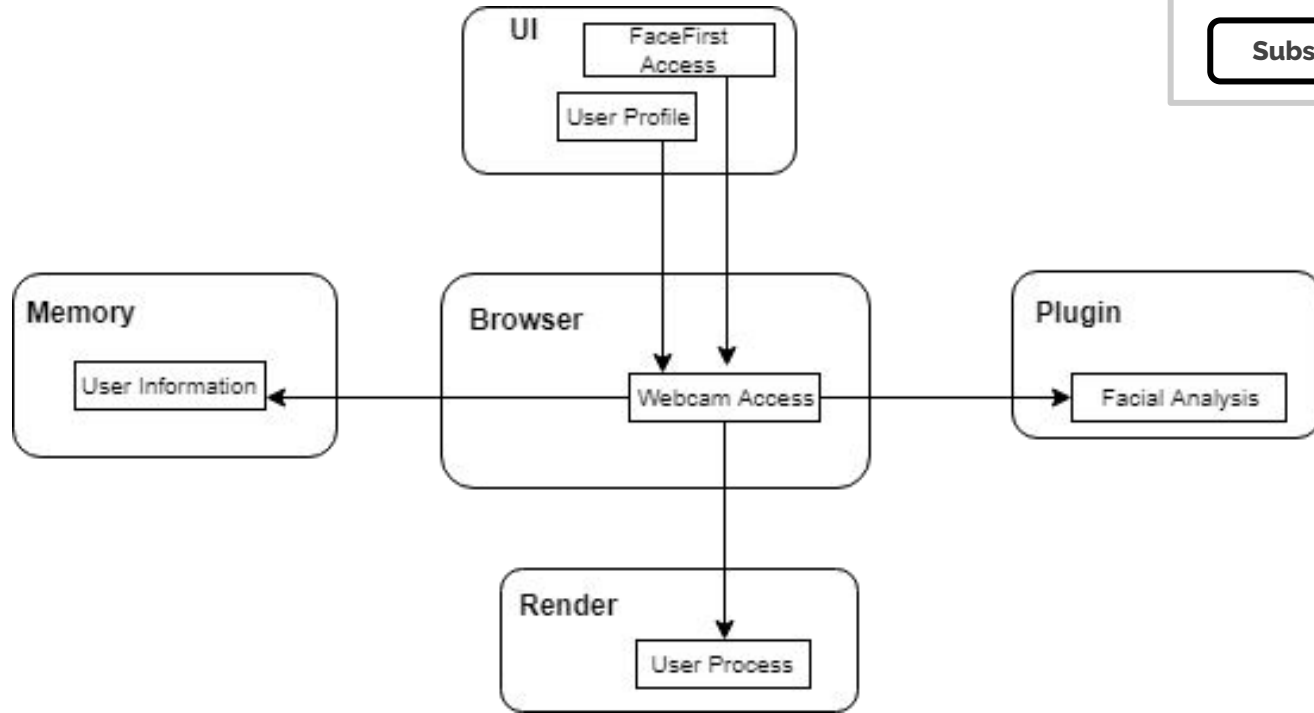
- Lower security
- Plugin not used



2nd Approach

- Facial analysis performed for each new tab that is opened, in addition to new browsers
- Constructed as a plugin for the browser
- Automatically logged out when tab is closed

2nd Approach



SAAM Analysis



Stakeholders

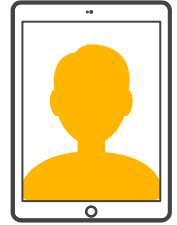
- Users
- Developers
- Investors
- Google

Pros

- Security
 - New login required for each new tab opened, ensuring same user sees content

Cons

- Speed
- More subsystems involved, higher coupling



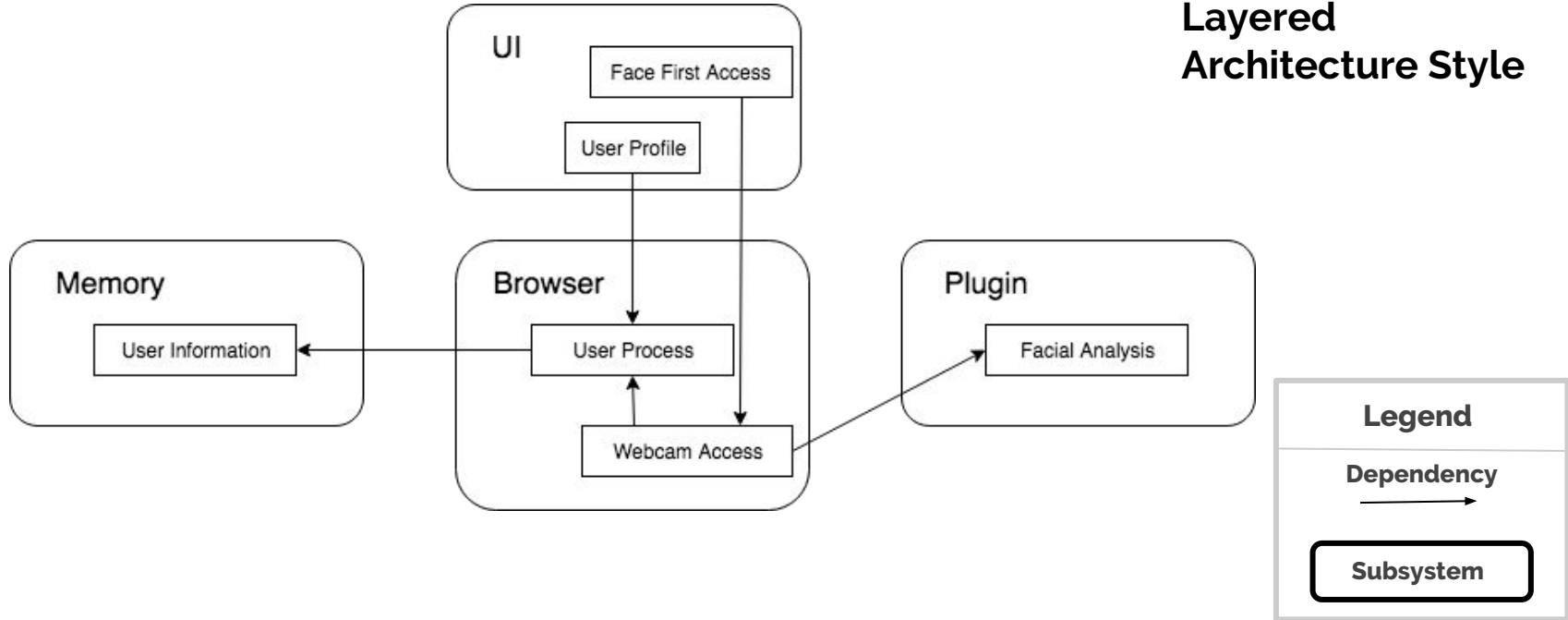
Final Approach

- Facial analysis performed in a plugin
- User data persists for all tabs in a window in user process
- Browser controls webcam access



Final Feature

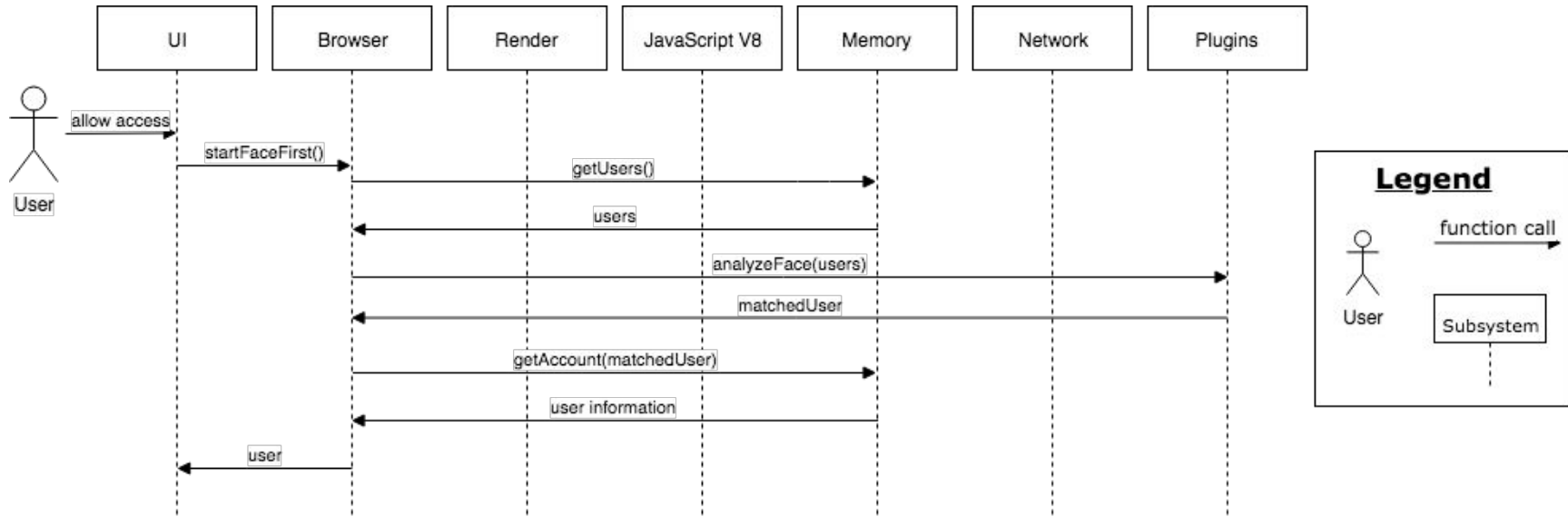
Layered Architecture Style





Sequence Diagram:

User wants to use Face First



Impacted Subsystems



User Interface

- UI will display a request for the user to login on browser start up
- UI sends request to the browser component to facilitate the Face First authentication
- UI dependencies remain unchanged from original architecture

Impacted Subsystems



Browser

- Browser receives authentication request from UI
- Browser accesses system's camera to capture the face of the user
- Browser requests original ID of the user from the memory component
- The browser sends the captured and original ID's to the plugin component for comparison



Impacted Subsystems

Memory

- Memory handles the browser request and returns original ID of the user
- User specific memory is restricted until the user is authenticated

Plugin

- The plugin component compares the captured and original ID's to validate the user.



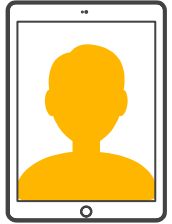
Risk and Limitation

Risk - Security

- Camera access in the browser component is a potential security risk from malicious applications

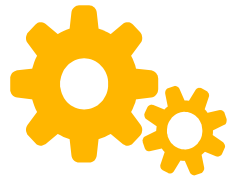
Limitation

- The camera quality of the system must be sufficient enough to perform analysis
- The camera must be functional



Testing Implementation

1. Test the feature's effect on different OS
2. Test that the existing password manager is not compromised
3. Test that the browser speed is not compromised
4. Test Face First functionality



Chrome's Concurrency Model

Multiprocess Architecture

- Each tab or plugin has its own process separate from the browser
- Helps protect against rendering failures

Supports multi-threading

- Main Thread
 - Browser Process: updates UI and processes the face recognition
- IO Thread
 - Browser Process: handles the IPCs which would include passing information to Memory and Plugins



Current Limitations and Lessons Learned

Current Limitations

- Did not implement, so we are unsure exactly how everything would interface and did not have a ton of reference

Lessons Learned

- Communication
- How to implement and integrate something completely new



Team Issues within Chrome

- Be aware of new security features
- Various new dependencies
 - Goal: reduce coupling



Conclusion

- After SAAM analysis, we determined our final approach using elements from both derivations
- Impacted subsystems include UI, Browser, Memory and Plugin
- Learned the steps to begin the implementation of a new product



Thanks!

Any questions?