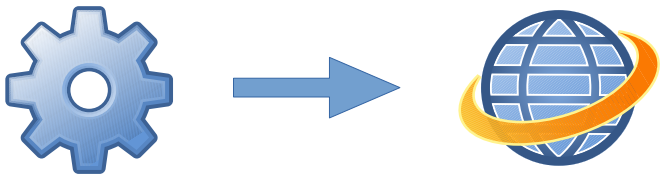
Parthorama

Our Intranet complements the Bricks and Mortar Clubhouse Model, benefiting the Psychocial Rehabilitation Community



The Problem

Bricks and mortar clubhouses foster face-to-face interactions,
but a virtual community is open 24/7
and is as close as your phone

<!>

To build a traditional virtual community,
web programmers are needed to cope with the complexity of
HTML, JavaScript and server-side code

<!>

A Facebook group is an inadequate, one-size-fits-all
solution to the task of building advanced virtual communities

The Solution

The Parthorama Intranet offers limitless ability to design the advanced functionality demanded by Progress Place and its members

<!>

Parthorama is implemented using Parthonyte and Parthotags, which can be accomplished by AI LLMs

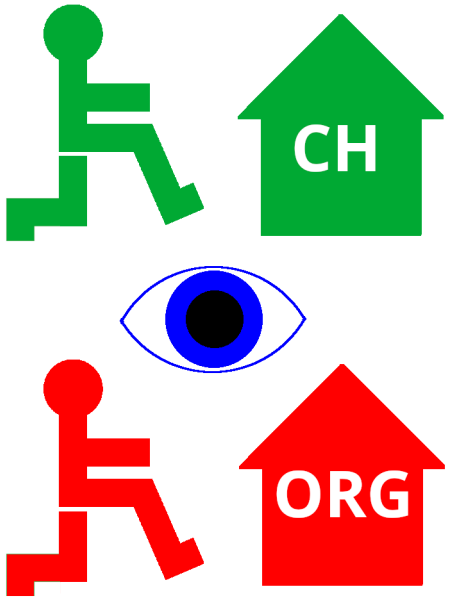
<!>

All development is fully under the control and guidance of Progress Place staff, no need to hire programmers and training costs are minimal, input from members can easily take place at all steps of the development process

Target Market

Green • Clubhouses
Green • Members
Mainstream Users:

- Red • Organizations
- Red • Employees
- Blue • Customers



Business Model

App authors who charge user fees pay royalties of 10%
plus 20% of ad revenue,
Organizations host sites and pay
$2.50/customer conversion,
plus $10/employee/year,
Nonprofit organizations pay no fees,
All organizations pay fees to
third party Java hosting services

Roadmap

1. Finish Parthonyte compiler
2. Integrate it with Parthotags
3. Launch website
4. Hire co-founder
5. Search for angel investor
6. Develop tool used for teaching math
7. Design intranet for Progress Place
8. Port system to Android
9. Port system to iOS (convert to Swift)

Timeline

- Foundation of Parthonyte code execution is complete
- **[6]** Develop rest of Parthonyte code execution: WCNMIL
  - Wrap up core foundation features
  - Classes and objects
  - Non-scalar data types
  - Modules
  - Inheritance + Interfaces (hedrons)
  - Library
- Release Parthonyte as console-based compiler on GitHub
- Begin recruiting contributors
- **[6]** GUI
- Write Parthotags design specs
- Develop Parthotags
- Integrate Parthonyte with Parthotags
- PYRE: Parthonyte Runtime Environment (open source)
- Develop Parthonyte code editor
- Expand code editor to Parthonyte SDK
- Launch website
- **[6]** Whiteboard used for teaching math
  - Variable width fonts (same time)
- **[3]** Develop online community for Progress Place

Total: 6 + 6 + 6 + 3 = 21 months

- **[9]** Develop mobile support:
  - Port PYRE to Android
  - Convert Android PYRE to Swift
  - Port Swift codebase to iOS
- **[3]** Lower priority features:
  - Implement Keyboard Aid (bells and whistles of editor)
  - Develop WYSIWYG Parthotags screen editor

Subtotal: 9 + 3 = 12 months
Grand Total: 21 + 12 = 33 months

Marketing

- Recruit Progress Place members
  - Expand to other clubhouses
- Use Google AdWords to reach new users

Team

1. Recruit volunteer coders and testers to help develop open source Parthonyte compiler
2. Team up with co-founder
3. Hire Android/Swift programmer to support mobile devices
4. Hire Chief Clubhouse Officer: liason with Progress Place, helps develop online community of members
5. Hire paid moderators
6. Hire CEO if necessary, I’d be the Chief Technology Officer

Revenue

Assume 1000 client-server apps, 1000 organizations, 1000 tutors exist.
Assume each app has $5K user fees, $5K ad revenue. Assume each site has 10 employees and 200 conversions. Assume each tutor has 4 learners.
Revenue = 1000(500 + 1000) + 1000(100 + 500) + 1000(60) = 1000(1500 + 600 + 60) =
$2.16 million.

Competition

Facebook has 2 billion users and very deep pockets, but clubhouses (helping consumer/survivors) want to support a builder of social networks which is not a giant corporation.

Secret sauce: build client-server software using world's simplest programming language.

Parthonyte and Parthotags

Parthonyte is implemented in Java and is Python-like, using parentheses for all grouping and operators come before their operands. Parthotags, a text markup language, is simpler than HTML, using square brackets. The client-server apps are built using these 2 languages.