

# **ITU-TRCSL Training on ICTs for promoting Innovation & Entrepreneurship**

## **A New ICT Education Framework**

**12-15 September, 2017  
Colombo, Sri Lanka**

**Pete Argent , ITU Expert**



## Agenda

- **Case Study: Technology Entrepreneurship Training in Sydney, Australia**
- **Modern Course Design Methodologies**
- **Attracting new students**





## About me

- IBM Software Consultant for 10 years
- Entrepreneur at heart
- Started teaching coding 2013
- Love teaching and giving people the power that comes with creating technology.





# Technology Entrepreneurship Training in Sydney, Australia

A case study





## Case Study

- Student demographics
- Student motivations
- A holistic approach: From idea to launch





# Student demographics





# Student demographics

- What types of people are interested in learning how to build their own technology enterprise?
  - Professionals
  - Recent university graduates
  - From all industries
    - Police
    - Doctors
    - Accountants
    - etc
  - Ages 14 – 60+
  - Men 60% Women 40% (increasing)





## Student motivations

- Want to learn coding to help advance their career
- Better understand technical colleagues when discussing projects. Have the language to communicate their ideas.
- Build their own tech startup. They can't afford to pay a developer, find a tech co-founder, so they want to learn enough to be able to build their own MVP.







## Incubators

- Incubators are organisations that help people build and launch tech startups.
- Most focus on teaching founders to pitch to investors with little or no meaningful education.
- Founders are left with limited understanding of what goes into the development of a technology product and are less likely to have a future-proof product.





## A new type of incubator

- Digital Intelligence believes the more the founder knows, the more chance they will build a successful startup.
- Our founders design, develop and launch their own startup through the program.
- Learning to code gives the founder a solid understanding of technology and create a better product as a result.





## A new type of incubator

- 6 month program – part time (2 evening classes a week)
- Homework challenges
- Step by step process from idea to launch
- Buddy system for added support and accountability
- Online platform with tutorials and a workbook for founders to work through during the program.





# The program

#	Module	Time
1	Startup Design	3 weeks (18 hours)
2	Web Design	3 weeks (18 hours)
3	Coding Fundamentals	4 weeks (24 hours)
4	App Developer	6 weeks (36 hours)
5	App Builder	4 weeks (24 hours)
6	Startup Launcher	5 weeks (30 hours)





## Startup Design

- Learn startup methodologies, human centered design, user experience.
- Students refine their startup idea and build a solid plan for the development of their startup.





## Web design

- Students learn HTML, CSS and Bootstrap
- Students build a landing page for their startup business





# Coding Fundamentals

- Students the fundamentals of coding with Ruby
- Become comfortable with programming concepts
- Introduction to web application development frameworks with Ruby on Rails





## App Developer

- Students build 12 apps over 6 weeks
- Each app is modelled on a real startup/online business like AirBNB, Facebook etc.
- Students become confident build any type of functionality and are prepared to build their own tech startup.







## App Builder

- Students get master classes for topics such as deployment, app optimisation, SEO etc
- Students build their own MVP with the support and guidance from teachers





## Startup Launcher

- Students plan their launch at a Demo Day event.
- Find beta testers for their product to iron out bugs and refine the product.
- Students present their startup and MVP to investors and families.





# Course Design Methodologies





# Modern Course Design

- Dealing with the fast pace of change
- Traditional vs Modern
- Methodologies





## Fast pace of change

- Technology is improving and changing every day
- If teachers have trouble keeping up – how will students be able to?





## Fast pace of change

- Technology is improving and changing every day
- If teachers have trouble keeping up – how will students be able to?
- **ONE SOLUTION:** Teach students how to teach themselves *or* “Learn how to learn”.





# Augmented intelligence

- No one can know everything
- Knowing how to find the information you need is important
  - Search skills & recognising quality articles/tutorials/websites
  - Reading and Understanding documentation
  - Using Q&A websites like StackOverflow and Quora
  - Finding and following tutorials
  - Keeping abreast of news, have an interest in what is going on outside your bubble
  - Teaching others helps you learn





## Traditional vs Modern Course Design

- In some industries, qualifications are becoming less important than demonstrable experience.
- Traditionally creative industries like music, art and design have long been like this.
- It is now being seen in programming.







## Traditional vs Modern Course Design

- Teaching students the information to pass exams is being seen as an inferior methodology.
- Practical, real world projects that prepare students for the work they will be doing can produce superior outcomes.





## Outcomes that impact peoples' lives

- In Australia, Universities seem to only care about their students graduating – not whether they have the skills required to secure a job.
- This can sometimes be due to the academics having limited real world work experience and not keeping up with changing technologies.





## Outcomes that impact peoples' lives

- Our goal should be to focus on delivering impactful outcomes for students.
- To have started an online business that has a growing customer base.
- To have a well paid job in their chosen career path.





## Teaching methodologies

- Quality course design, development
- Passionate teachers
- Buddy system
- Gamification
- Built in peer support systems
- Experts and mentors
- Feedback loops & continual improvement





# Quality course design and development

- Consider
  - situational factors
  - impactful outcomes
  - themes of course
- Plan activities that challenge students to discover information and skills on their own and in teams.
- Less lecturing and more discussions – no spoon feeding of information





## Real world projects

- Challenging students to find a real world project to work on is powerful.
- We ask students to find a local business owner who needs a website.
  - Experience finding and working with a client
  - Helping local community
  - Adds to student's portfolio
  - Gains confidence





## Passionate teachers

- We all recognise that a teacher's role in class is not just to impart information.
- It is our job to inspire our students to want to learn, to work hard and be successful.
- Learning should be fun! If we focus on making sure our students are having fun, we have fun too!





## Buddy system

- In some classes it is a great idea to have a buddy system.
- Students form pairs to provide additional accountability and support.
- Higher achievement by all.







## Gamification

- A great tool to encourage participation and adoption of personal qualities that are preferred by employers.
- We give award points for:
  - Quality code
  - Agility
  - Collaboration
  - Leadership
  - Community participation
- A quality prize and recognition is important.





## Peer support systems

- Chat boards like Slack
- Peer review projects and activities
- Students learn from their peers as much as from the teacher and encourages participation and collaboration.





## Experts and mentors

- Bringing in people who work in the industry allows students to connect and network and hear about different companies.
- Shows them the teacher knows what they're talking about 😊
- An excursion to a workplace is a bonus.





## Feedback and continual improvement

- Tight feedback loops – don't wait til the end of the course to get it.
- One-on-one sessions
- Update course materials regularly





# Attracting new students





## Attracting new students

- Some people already know they need to learn about technology to improve their career or start a technology focused business.
- We aim to reach people who don't realise the opportunities that are available to them from learning technology skills.





## Activities to attract new students

- Free workshops
- Hackathons
- Meetup groups
- Primary school & high school visits
- Community events
- Speaking engagements
- Videos and podcasts





# I Thank U (ITU)

