# ICT-Skills Studio (2016)

## Higher Diploma in Science in Computer Science

---

Produced by

John Fitzgerald (jfitzgerald@wit.ie)

Department of Computing, Maths & Physics
Waterford Institute of Technology

http://www.wit.ie

http://elearning.wit.ie

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

eLearning support unit

# Assignment

Web Development

# ICTSkills Studio 2016

## Assignment Rubric

| Description | | Grade (%) |
|---|---|---|
| Assignment | MyRent* | 85 |
| Git application | Donation | 7.5 |
| | WitPress | 7.5 |
| | Total | 100 |
| | | |
| * | Web app | |
| | Git application | |
| | Live demo | |
| | Interview | |
| | User experience (UX) | |
| | Developer Experience (DX) | |
| | | |

# Assignment requirements

- For each lab web app, Donation, MyRent and WitPress, create a local git repository (repo).

- Commit often (with suitable messages).

- Tag iterations using tag names- for example Donation: v0, v1 ..v4.

- Use similar naming for MyRent and WitPress tags (v0, v1 and so on).

- Track your local git repo with a cloud repo on BitBucket.

- Provide a link to the cloud repo as part of your assignment submission.

- Share your repos with your tutor(s): Bitbucket ids to be provided.

# Assignment requirements (continued)

- Continue with the MyRent repo in developing your assignment project.

- Commit (with suitable messages) each story and intermediate steps.

- Apply a tag to each committed story: story-01, story-02 and so on.

- Provide tag messages that are short but descriptive of the story.

- Use Semantic UI CSS Framework.

- Your final MyRent repo should have 2 branches: master (configured for localhost) and heroku (cloud).

- Deploy to the cloud (Heroku) the MyRent web app project you submit as your assignment.

# Assignment requirements (continued)

- You may make as many commits as you wish but only the specified tags may be applied. These are, in summary:

  - The iteration tags for the labs, for example v0, v1, v2 and so on.

  - The story tags for the assignment, for example story-01, story-02 and so on. **All** tags in all apps lower case please.

- Stories may not be skipped. For example, if your goal is to achieve a *Good* grade then you are required to complete all of stories 1 to 10 inclusive. This applies even where some previous code may become obsolete in the refactoring.

- The application should compile and run without error for each story.

# Stories - Baseline

- *Story 1:*
  - Provide for additional Residence fields
    - *int numberBathrooms;*
    - *int area;* // the area of the residence in square metres
  - Refactor the input data template (html) page appropriately.

- *Story 2:*
  - Report to contain extra columns showing new Residence fields.
  - Include new fields in data.yml.

- *Story 3:*
  - Rename model User to Landlord, rename controller Accounts to Landlords and refactor all dependencies appropriately.

- *Story 4:*
  - Add a landlord Edit Profile component with menu access to logged-in landlord from input data page. Allow edit of name and address.

# Stories - Baseline (continued)

- *Story 5:*
  - Add a Tenant model and associated controller (Tenants) and views. Provide sign up and login features. Tenant fields to be firstName, lastName, email, password and Residence reference. Implement authorisation.
  - The welcome or home page should now contain three fully implemented menu items: Home, Landlord, Tenant. The latter two should lead to login pages for Landlord and Tenant respectively.
  - Refactor the data.yml file, ensuring it contains sample Residence,

# Stories - Good

- *Story 6:*
  - Add Landlord configuration page comprising Add Residence and Edit Landlord Profile buttons, Delete Residence and Edit Residence dropdown boxes.
  - The Add Residence button press switches to Input Data page (existing).
  - Edit Landlord Profile switches to Edit Profile page (existing).
  - Populate dropdowns with list residences. Actions implementations later.

- *Story 7:*
  - Implement Delete Residence.
  - Implement Edit Residence. Allow editing of rent field only. Display Eircode as readonly field.

- *Story 8:*
  - New model Residence fields:
    - *Landlord landlord*: one to many. Landlord may own many residences.
    - *Tenant tenant*: one to one. One tenant per residence and vice versa.
  - Update yml file with additional sample data.

# Stories - Good

- *Story 9:*
  - Residence field:
    - The residence field *String rented* is redundant as this state is captured by value of the Tenant reference. Delete the redundant field.
    - Remove the corresponding radio button pair in the Input data page.
    - Fix any anomalies that may arise and modify dependencies appropriately.

- *Story 10:*
  - Logout:
    - Use a separate Session scope to maintain a record of each logged-in user - Landlord or Tenant.
    - Implement a logout action for each of the users.
    - Add a menu entry with appropriate routing.

# Candidate Stories - Excellent

- *Story 11:*
- On successful authorisation a tenant should be routed to a page with these features:
  - Readonly eircode representing existing rental.
  - Button to terminate tenancy.
  - Dropdown to allow selection of new tenancy from list vacant residences.
  - Googlemap-based vacant residences search module.
- Fully implement the report (search result) + model, view & controller code.

- *Story 12:*
  - Implement an Administration module comprising:
    - Administrator model containing hardcoded email & password
      - Use: *admin@witpress.ie, secret*
    - Login in page with corresponding Administrators controller authorisation. Add a logout feature also to all Admin pages.
    - On login, switch to administration page containing:
      - Button to register (signup) landlord.
      - Button to register (signup) tenant.
      - Google map representing all residences with markers + tooltips stating eircode & tenant.

# Stories - Excellent

- *Story 13:*
  - Administration page - add:
    - Button to delete tenant.
    - Button to delete landlord.
  - Refactor map, model, view & controller code as appropriate to reflect any Landlord or Tenant deletions.

- *Story 14:*
  - Implement Administrator reporting module. The html (template) page should include filters to generate reports by rented status (vacant or rented), by type (house, flat and so on), by amount rent sorted ascending or descending and the unfiltered list.
  - The columns should include eircode, date residence registered, type, number bedrooms, number bathrooms, rent (€), area, landlord name, rented status and tenant if any.

- *Story 15:*
  - Input data page - add:
    - Facility to manually obtain eircode correspoding to GPS (lat, long) using EirCODE Finder.

# Stories - Outstanding

- *Story 16:*
  - Implement a charting module accessible to the logged in administrator and comprising:
    - Table of residences data: landlord, eircode & rent.
    - Pie chart representing landlords total rent roll as a percentage of the entire database residence portfolio income.

- *Story 17:*
  - Implement Semantic UI validation on all input controls. Use external JavaScript files and locate them in *public/javascripts folder.* Note that the validation approach has changed from that used in Semantic version 1.
  - Apply airbnb styling to JS files using WebStorm.

# Stories - Outstanding

- *Story 18 (Ajax):*
  - *Administration page*
    - Implement jQuery ajax for:
      - Delete Landlord
      - Delete Tenant
      - Changes to map resulting from either above deletion.
    - Dynamically adjust dropdowns' content to reflect deleted tenant or landlord.

# Stories - Outstanding

- *Story 19 (Ajax):*
  - *Tenant page*
    - Implement jQuery ajax for:
      - Terminate Tenancy
      - Change Tenancy
      - Changes to map resulting from either above deletion.
    - Adjust dropdown content to reflect changed vacant residence list (eircodes).

- *Story 20 (Ajax):*
  - *As story 11 & 19 except that the search area in the map is defined by a polygon rather than a circle.*
  - *Hint: Mine the Polygon Map Overlay lab and associated JavaScript VI presentation (week 4) for relevant material.*

# story-01

Number bathrooms  2

Area (sq. m):  110

```
@Entity
public class Residence extends Model
{
  …
  public int numberBathrooms;

  public int area;
}
```

# story-02

| Welcome | Sign Up | Log In | Input data | Report | Log Out | Contact |
|---------|---------|--------|------------|--------|---------|---------|

**Report**

| Commissioned by | Search: radius(m) | centre(latitude) | centre(longitude) |
|-----------------|-------------------|------------------|-------------------|
| Homer Simpson | 191777.2 | 53.34730 | -6.26834 |

| Residence Id | Residence User | Date registered | Rented? | Rent(€) | Bedrooms | Bathrooms | Area |
|--------------|----------------|-----------------|---------|---------|----------|-----------|------|
| 1 | Homer Simpson | 05 June 2016 | yes | 1500 | 3 | 2 | 110 |
| 2 | Homer Simpson | 05 June 2016 | yes | 2500 | 1 | 1 | 90 |

# story-03
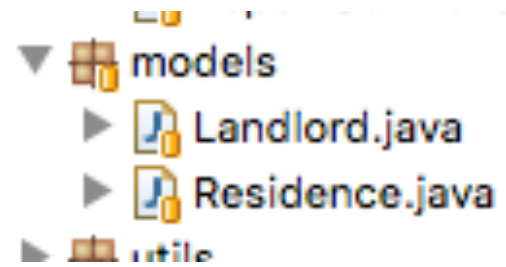
# story-04

| Welcome | Sign Up | Log In | Input data | Report | Edit Profile | Contact |
|---------|---------|--------|------------|--------|--------------|---------|

## You may edit your profile here

**First name**

Homer

**Last name**

Simpson

**Address Line 1**

address1

**Address Line 2**

address2

**City**

city

**County**

county

[Submit] [Cancel]

# story-05

## Model

- ▼ Tenants.java
  - ▼ Tenants
    - authenticate(String, String) : void
    - getCurrentTenant() : Tenant
    - index() : void
    - login() : void
    - logout() : void
    - register(Tenant) : void
    - setSessionLogout() : void

## View

- ▼ Tenants
  - index.html
  - login.html

## Controller

- ▼ Tenant.java
  - ▼ Tenant
    - findByEmail(String) : Tenant
    - email
    - firstName
    - lastName
    - password
    - residence
    - checkPassword(String) : boolean
    - toString() : String

---

| Welcome | Sign Up | Contact |

### Tenant registration

**Sign Up**

First Name
Barney

Last Name
Gumble

Email
barney@gumble.com

Password
••••••

**Submit**

### Log into your tenant account

Your username is the email you registered with

**Log-in**

Email
barney@gumble.com

Password
••••••

**Login**

20

# story-06

Home

**Landlord configuration page**

| Add Residence | Edit Landlord Profile |

| Select residence to delete ▾ | Edit Residence ▾ |

| Delete residence | Edit residence |

# story-07

**Landlord configuration page**

**Add Residence**

**Edit Landlord Profile**

Select residence to delete ▾

Edit Residence ▾

**Delete residence**

**Edit residence**

# story-07

Home

**Please update residence data**

| Eircode | X82 X999 |

| Monthly rent (€) | 2000 |

**Update data**

# story-08

```
@Entity
public class Residence extends Model {

        @ManyToOne
        public Landlord landlord;

        @OneToOne(mappedBy = "residence")
        public Tenant tenant;
```

```
@Entity
public class Landlord extends Model
{
         @OneToMany(mappedBy="landlord", cascade=CascadeType.ALL)
         public List<Residence> residences = new ArrayList<Residence>();
```

```
@Entity
public class Tenant extends Model {

        @OneToOne
        public Residence residence;
```

```
Landlord(homer):
    firstName: Homer
    …
    email: homer@simpson.com
    password: secret
    residences: [residence1, residence2, residence3]
```

```
Tenant(barney):
    firstName: Barney
    lastName: Gumble
    email: barney@gumble.com
    password: secret
    residence: residence1
```

```
Residence(residence1):
    geolocation: "53.220565, -6.659308"
    eircode: "W91 X934"
    dateRegistered: "2015-05-23 16:53:08.543"
    rent: 1750
    …
    landlord: homer
    tenant: barney
```

# story-09

# story-10

| Welcome | Sign Up | Log In | Input data | Report | Log Out | Contact |
|---------|---------|--------|------------|--------|---------|---------|

```java
public static void logout()
  {
    setSessionLogout();
    Home.index();
  }
```

# story-11

Home　　Tenant　　Contact　　Log out

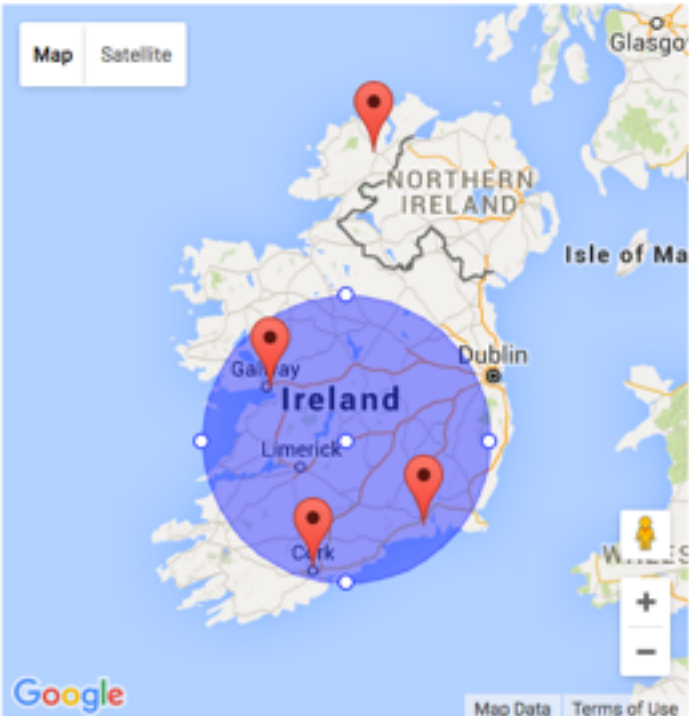**Existing rental (eircode)**

W91 X934

**Press to end tenancy**

[ Terminate tenancy ]

**Select from vacant residences for new rental**

Vacant residences ▼

[ Change Tenancy ]



Map　Satellite

[ Search for vacant residences ]

**Vacant residences**

| Tenant | Search: radius(m) | centre(latitude) | centre(longitude) |
|---|---|---|---|
| Barney Gumble | 118845.8 | 52.85898 | -8.06776 |

| Eircode | Date registered | Rent(€) | Bedrooms | Bathrooms | Area (m²) |
|---|---|---|---|---|---|
| U94 D922 | 23 May 2015 | 2500 | 4 | 2 | 100 |
| V94 D922 | 23 May 2015 | 2500 | 4 | 2 | 100 |
| H91 X28D | 23 May 2015 | 2500 | 4 | 2 | 100 |

# story-12

# story-13

## Administration page

### Landlords and Tenants

**Register landlord**

**Register tenant**

Select landlord to delete ▾

**Delete landlord**

Select tenant to delete ▾

**Delete tenant**

**Single click on marker to read eircode and tenant**
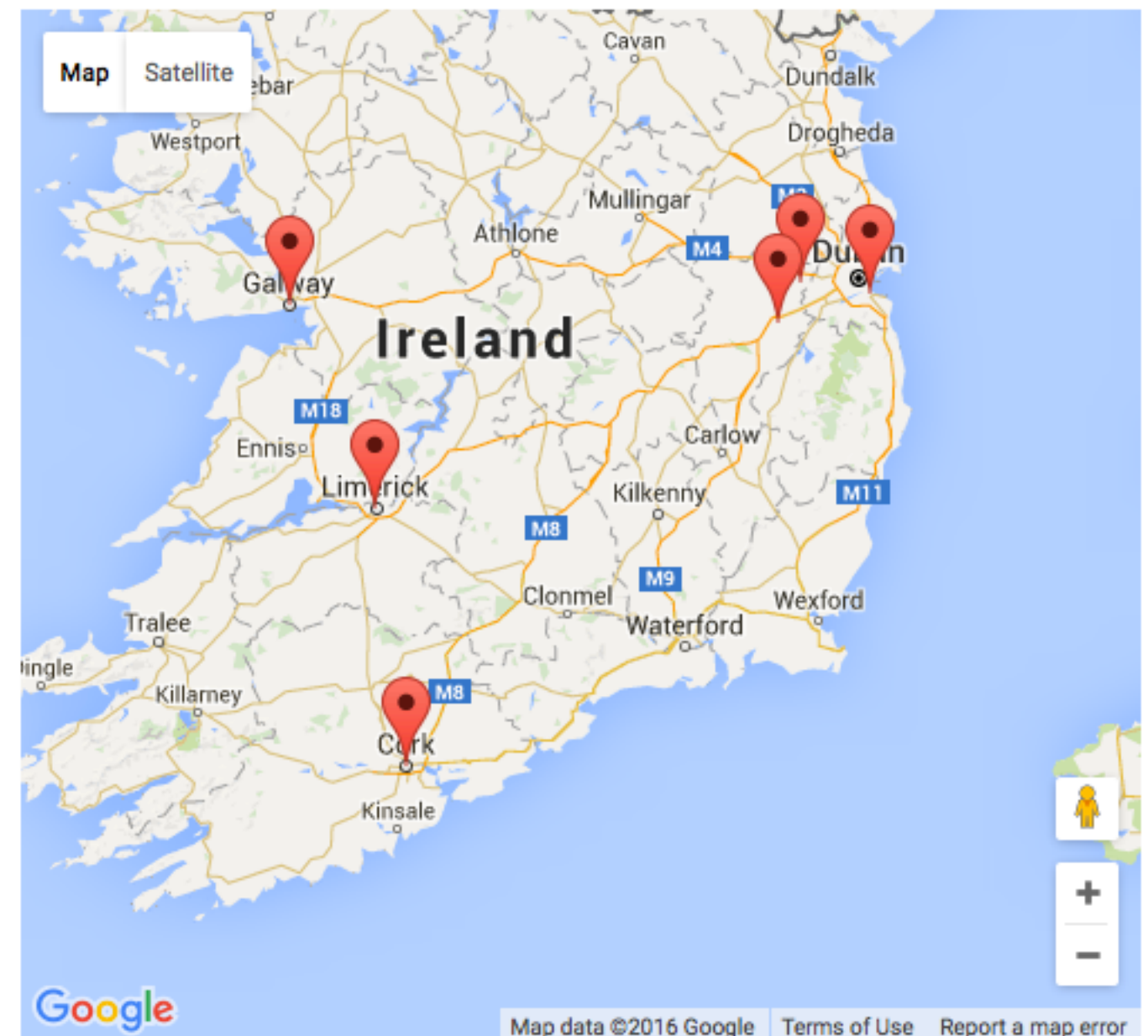
# story-14

**Report filtering**

| Eircode | Date | Type | Number bedrooms | Number bathrooms | Rent($) | Area (m$^2$) | Landlord | Tenant |
|---------|------|------|-----------------|------------------|---------|--------------|----------|--------|
| W91 X934 | 23-05-2015 | house | 3 | 2 | 1750.0 | 80 | Homer Simpson | Barney Gumble |
| A94 X439 | 23-05-2015 | house | 4 | 2 | 2000.0 | 100 | Homer Simpson | Klovn Krasti |
| W23 X433 | 23-05-2015 | flat | 4 | 2 | 800.0 | 50 | Homer Simpson | Krusty Clown |
| T12 X466 | 23-05-2015 | studio | 4 | 2 | 750.0 | 45 | Marge Simpson | Ned Flanders |
| H91 W7Y2 | 23-05-2015 | flat | 4 | 2 | 2000.0 | 100 | Mr. Withers | Moe Szyslak |
| V94 D922 | 23-05-2015 | house | 4 | 2 | 2500.0 | 100 | Mr. Withers | |

By Rented **Filter**    By Type **Filter**    Rent (€) **Filter**    **All**

# story-15
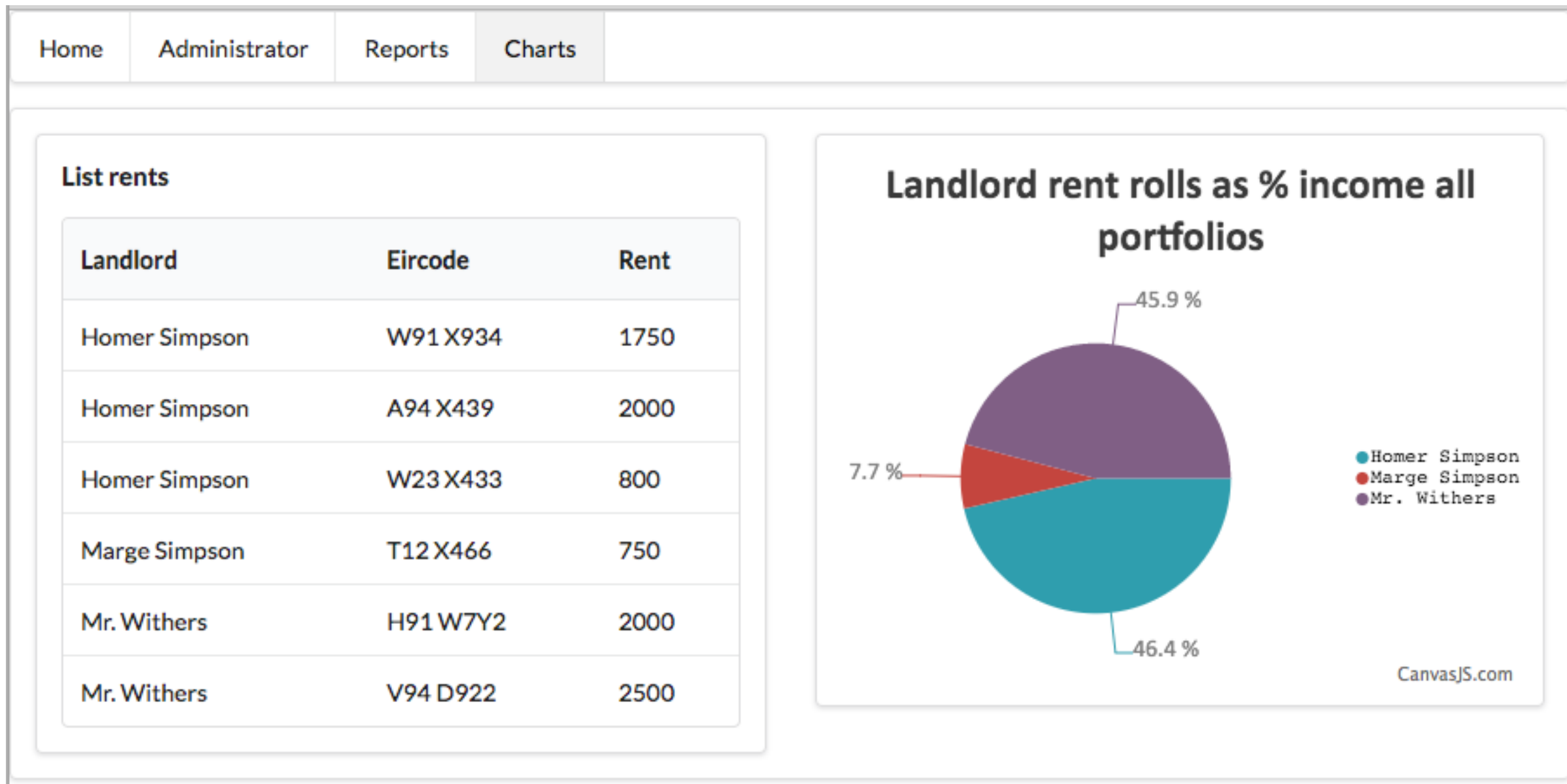
# story-16

# story-17

Semantic UI Validation

```
$('#eircode_residence').dropdown();

$('.ui.form.changeTenancy').form({
  fields : {
    eircode_vacancy : {
      identifier : 'eircode_vacancy',
      rules : [{
        type : 'empty',
        prompt : 'Select a vacant residence (if available) before pressing button'
      },],
    },
  },

  onSuccess : function(event, fields) {
    changeTenancy();
    event.preventDefault();
  },
});
```

# story-18

Example: on deleting landlord the landlord name to be dynamically removed from dropdown, the map markers and tooltips to be updated (removed, edited …), all using jquery ajax. Same applies for tenant

| Home | Administrator | Reports | Charts | Contact | Log out |

**Administration page**

**Landlords and Tenants**

Register landlord

Register tenant
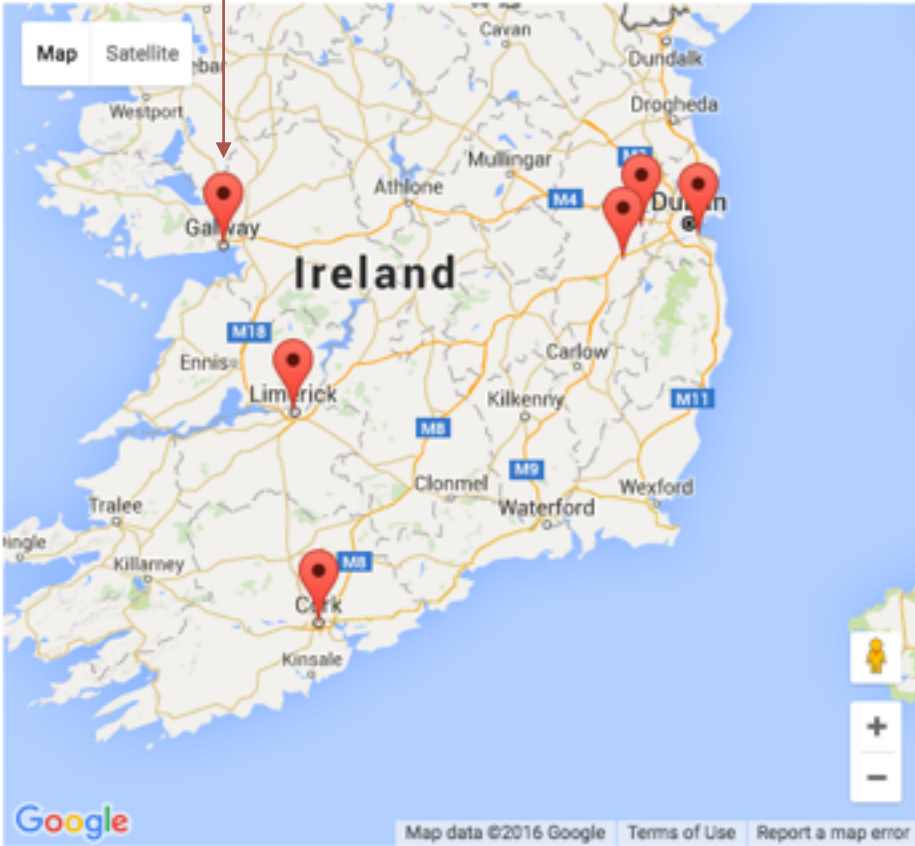
Select landlord to delete ▼

homer@simpson.com

marge@simpson.com

mr@withers.com

Select tenant to delete ▼

Delete tenant

Single click on marker to read eircode and tenant

# story-19

Map markers represent vacant residences. Update using ajax to reflect tenancy changes

Home    Tenant    Contact    Log out

**Existing rental (eircode)**

W91 X934

**Press to end tenancy**
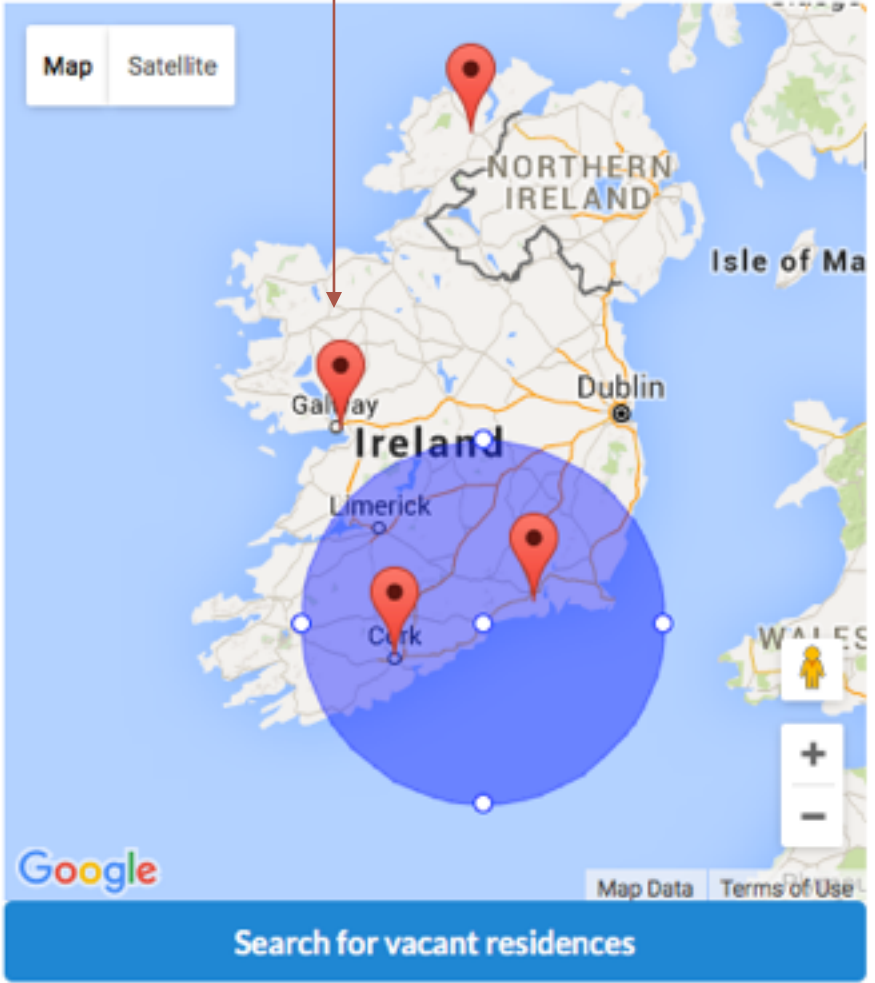
Terminate tenancy

**Select from vacant residences for new rental**

Vacant residences ▼

Change Tenancy

Map    Satellite

Dublin

Ireland

Limerick

Cork

WALES

Google

Map Data    Terms of Use

Search for vacant residences

Use ajax to dynamically adjust dropdown content

# story-20



Map markers represent vacant residences.
Update using ajax to reflect tenancy changes
Use polygon to define search region.

Home    Tenant    Contact    Log out

**Existing rental (eircode)**

W91 X934

**Press to end tenancy**
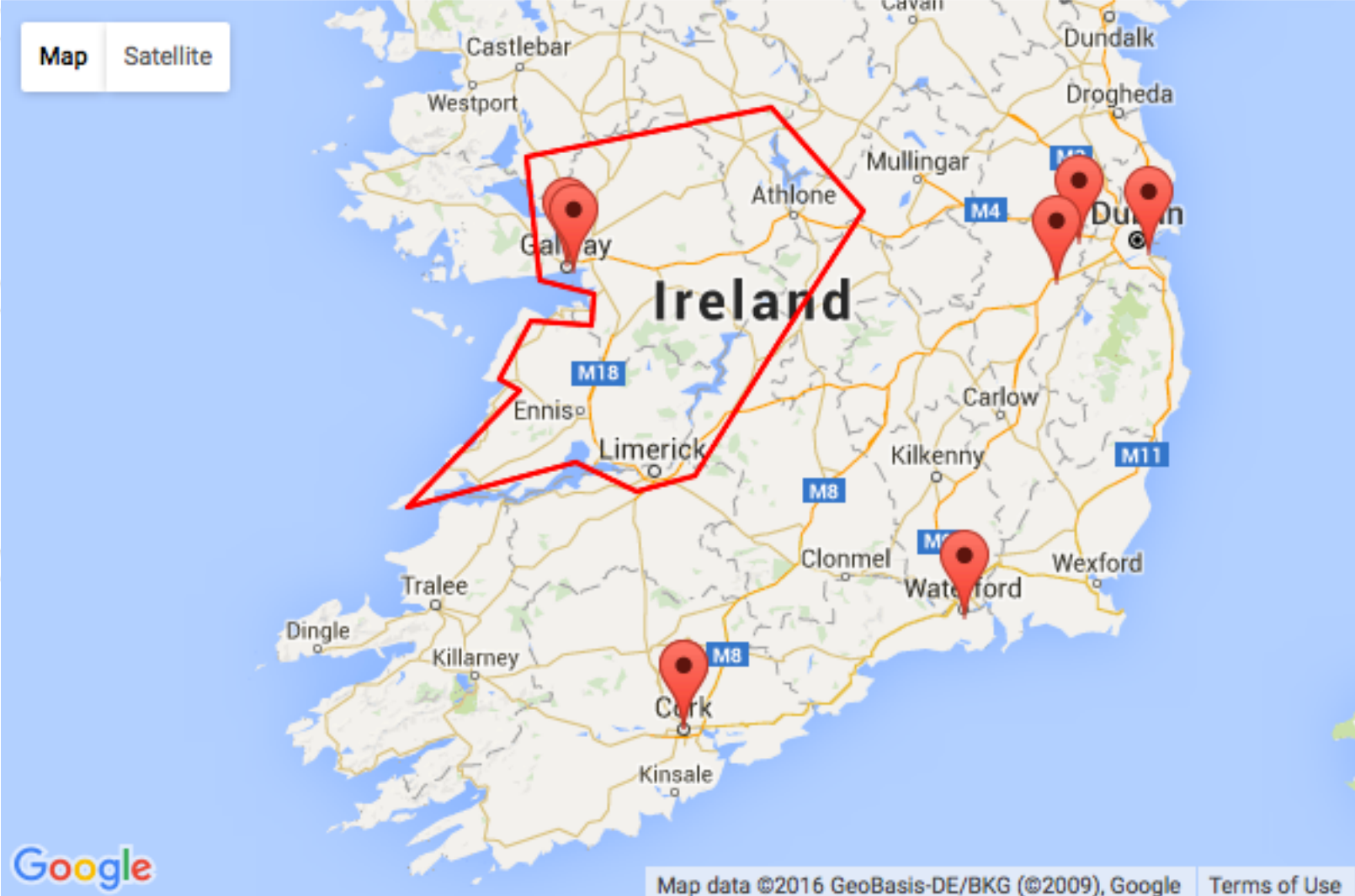
Terminate tenancy

**Select from vacant residences for new rental**

Vacant residences

Change Tenancy

Use ajax to dynamically adjust dropdown content

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

eLearning support unit